Meerwaarde van calibratie

Een gevalstudie

Sabrina Geboers

promotor :
Prof. dr. Koen VANHOOF

Eindverhandeling voorgedragen tot het bekomen van de graad

universiteit hasselt

Woord vooraf

Ter afsluiting van mijn studies Handelsingenieur in de beleidsinformatica aan de Universiteit Hasselt schreef ik deze masterproef.

Graag zou ik mijn promotor Prof. Dr. Koen Vanhoof willen bedanken voor de begeleiding, de raadgevingen en de suggesties die hij gedaan heeft. Ik kon steeds bij hem terecht voor de nodige uitleg en adviezen.

Ook wil ik mijn ouders en tweelingzus bedanken voor de steun die ik kreeg bij het maken van deze masterproef. Dankzij mijn ouders heb ik deze studie kunnen aanvatten en ze hebben mij steeds gesteund bij het behalen van mijn diploma.

Samenvatting

Data mining wordt de laatste jaren belangrijker, aangezien er een overvloed aan informatie bestaat. Met behulp van data-mining-technieken kunnen we de relevante informatie uit grote hoeveelheden gegevens halen. Er bestaan verschillende methoden die hiervoor gebruikt kunnen worden, zoals clustering, classificatie enz. De applicatie die in deze masterproef gebruikt wordt, is een applicatie die selecteert, dit wil zeggen dat er een set van meest winstgevende cases gevonden moet worden. Hiervoor wordt een classifier gebruikt. In deze masterproef wordt er gebruik gemaakt van probabilistische classifiers. Deze classifiers wijzen aan elke observatie een waarschijnlijkheid toe. Deze probabiliteit moet de ware waarschijnlijkheid dat een case tot de positieve klasse behoort, voorstellen. Om te weten welke classifier de beste oplossing geeft, moeten we de verschillende modellen gaan vergelijken.

De calibratietechniek, het Pool Adjacent Violaters (PAV) algoritme, genereert intervallen voor de verschillende classifiers en geeft aan elk interval een unieke probabiliteit mee. Elke case wordt op zijn beurt toegewezen aan één uniek interval. Een opvallend kenmerk is nu dat elk model wel intervallen heeft, maar het aantal intervallen en de posities van de intervallen niet dezelfde zijn. Dit bemoeilijkt de vergelijking tussen de verschillende classifiers. In deze masterproef wordt op zoek gegaan naar een nieuwe calibratiemethode, die gebaseerd is op verscheidene modellen.

Om het beste model eruit te halen, wordt er gebruik gemaakt van een klassiek criterium, namelijk de Brier score. Deze score kan opgesplist worden in calibration loss en refinement loss. De calibration loss geeft terug hoe goed het model de echte verdeling weergeeft. De refinement loss neemt het gemiddelde van de wanorde in de toewijzing van de probabiliteiten over de verschillende intervallen. Er bestaat een trade-off tussen deze twee elementen, het is moeilijk om beide componenten te optimaliseren.

Vervolgens worden er voor drie methoden, namelijk beslissingsboom, logistische regressie en discriminant-analyse, de Brier score en zijn componenten berekend. Dit zowel voor de gegevens waarop de calibratietechniek niet is toegepast, als voor de

gegevens met calibratie. Het doel is uiteraard om de verschillende modellen te vergelijken. Met behulp van de calibratietechniek zijn er al intervallen gegenereerd. Wat er daarna gedaan wordt, is het aantal intervallen reduceren, zodat elke classifier hetzelfde aantal heeft en bovendien de posities van deze intervallen op dezelfde plaats liggen. Deze vermindering van intervallen gebeurt op drie verschillende manieren in dit onderzoek. Bij de eerste manier maken we gebruik van een algoritme om gemeenschappelijke intervallen te verkrijgen. De tweede manier gebeurt door rekening te houden met de minimum refinement loss en tenslotte maken we gebruik van drie gecalibreerde probabiliteiten om de reductie tot stand te brengen.

Nadat deze berekeningen gemaakt zijn, wordt er getracht om de verschillende modellen met elkaar te vergelijken. Er wordt vastgesteld dat het vergelijken van deze classifiers nog altijd niet mogelijk is, ook al hebben we dezelfde intervallen en liggen de posities vast. De reden hiervoor is dat de verdelingen van de verschillende modellen sterk van elkaar afwijken. De verdeling van een model verschilt van een andere classifier, doordat er bij sommige intervallen geen cases voorkomen, waar dit interval bij de andere classifier bijvoorbeeld wel observaties bevat. Bovendien kan er geconcludeerd worden dat de cases niet gelijkmatig verdeeld zijn over de resterende intervallen. Zo heeft bijvoorbeeld één enkel interval de meerderheid van de observaties; dit komt vooral voor bij de discriminant-analyse.

Als belangrijkste conclusie uit deze masterproef kan worden vastgesteld dat niet enkel het aantal intervallen en de posities van deze intervallen van belang zijn, maar ook de toewijzing van de observaties aan deze intervallen. Indien er drie modellen bestaan met ongeveer dezelfde verdeling - dit betekent een gelijkmatige indeling in de intervallen - zouden we beter in staat zijn om deze classifiers te vergelijken; in elk interval bevinden zich dan met zekerheid een aantal cases.

Inhoudsopgave

Lijst van figuren

Lijst van tabellen

Hoofdstuk I : Inleiding

A. Probleemstelling

Data mining is een techniek die gebruikt wordt om uit enorme hoeveelheden gegevens, patronen en relaties te ontdekken en dit op een geautomatiseerde manier. De laatste jaren is er een overvloed aan informatie, waardoor data mining belangrijker wordt. Met deze techniek kan men de relevante informatie uit de data halen en dan optimaal benutten. Voor het zoeken naar deze kennis kan men gebruik maken van verschillende methoden zoals classificatie, clustering, associatieregels enz. De applicatie die in dit onderzoek gebruikt wordt, is een toepassing die selecteert. Dit wil zeggen dat er een set met de meest winstgevende cases in, gevonden moet worden.

Het is niet alleen belangrijk om de informatie uit de data te halen met een specifieke techniek, maar de prestatie van deze methode moet bovendien beoordeeld worden. Er bestaan verschillende grafische technieken, waarmee de kwaliteit van het classificatiemodel geëvalueerd kan worden. Vuk en Curk (2006) geven een aantal van deze methoden weer. Één daarvan is de receiver operating characteristic (ROC) curve. Dit is een grafiek die de ratio van de 'true positives' tegenover de ratio van de 'false positives' zet. Deze curve is uiterst geschikt voor het vergelijken van classifiers. Een andere grafiek, die zij bespreken, is de 'lift chart'. Lift is een maatstaf voor de effectiviteit van een voorspellend model. De lift wordt berekend door de verhouding te nemen tussen de resultaten die met en zonder het voorspellende model verkregen werden. De lift chart toont de werkelijke lift en wordt vooral toegepast bij marketingproblemen. Deze grafiek is een grafische voorstelling van het voordeel dat men krijgt door gebruik te maken van een voorspellend model, zodat men kan kiezen welke personen men gaat contacteren. De lift chart geeft weer hoeveel waarschijnllijker het is dat mensen gaan reageren dan als we willekeurige personen zouden contacteren. Als laatste bespreken de auteurs de calibration plot. De calibratiecurve is een methode die weergeeft hoe goed de classifier gecalibreerd is. Calibreren is het instellen en aanpassen van een systeem, zodat het voldoet aan de specificaties ten aanzien van nauwkeurigheid en betrouwbaarheid.

In supervised learning zou een model dat de echte onderliggende waarschijnlijkheid voor elke testcase voorspelt, optimaal zijn. Caruana en Niculescu-Mizil (2004) verwijzen naar zo een model als het enige echte model of 'the one true model'. Elk aannemelijk prestatiecriterium zou geoptimaliseerd moeten worden door het enige echte model en geen enkel ander model zou een betere prestatie mogen opleveren. Meestal weten we niet hoe we modellen moeten trainen, zodat ze de enige ware waarschijnlijkheden zouden voorspellen. Het enige echte model is niet makkelijk aan een computer aan te leren, aangezien het correcte modeltype met parameters voor het domein niet gekend is of omdat het aantal trainingscases te klein is om de modelparameters nauwkeurig te schatten. Het kan bovendien zijn dat er 'noise' of ruis in de data voorkomt. Ruis in de data wil zeggen dat er corrupte of foute gegevens in de dataset aanwezig zijn. Dit kan veroorzaakt worden doordat de metingen die gedaan werden, niet correct gebeurd zijn of doordat de input die er is, niet goed is. Gewoonlijk komen deze problemen tesamen voor, maar wel in variërende niveaus. Ook al zouden we het enige ware model krijgen, dan nog zou het moeilijk zijn om het te selecteren uit de andere minder echte modellen. In de praktijk zoeken we naar modellen waarbij het verlies dat via een specifiek evaluatiecriterium wordt gemeten, geminimaliseerd wordt. Aangezien er geen criteria bestaan die het enige ware model selecteren, moeten we accepteren dat het model dat we selecteren waarschijnlijk suboptimaal zal zijn.

Huang en Ling (2007) vermelden dat evaluatiecriteria veel gebruikt worden in machine learning en data mining om verschillende leeralgoritmes met elkaar te vergelijken. Bovendien worden ze toegepast als objectieve functies om leermodellen te construeren.

B. Centrale onderzoeksvraag

Het resultaat dat men bekomt door data-mining-technieken toe te passen, verschilt van methode tot methode. Elke classifier die men gebruikt, heeft een ander aantal intervallen en de posities van de eindpunten van deze intervallen liggen bovendien niet op dezelfde plaats. Vuk en Curk (2006) definiëren een classifier als een classificatiemodel dat voorbeelden indeelt in de correcte klasse. Aangezien het belangrijk is om een goed model, dat het meest suboptimaal is, te kiezen, is het dus ook van belang dat we de

modellen met elkaar kunnen vergelijken. Doordat het aantal intervallen varieert, is het moeilijk om een juist beeld te krijgen, als men de methoden met elkaar wil vergelijken.

Een onderzoeksvraag zou dan kunnen zijn: 'Wat is het effect en nut van een nieuwe calibratiemethode gebaseerd op meerdere modellen?'

De deelvragen die ik hierbij geformuleerd heb, zijn de volgende:

1. -Wat zijn de voordelen van deze nieuwe calibratiemethode?

2. -Wat zijn de nadelen/beperkingen van deze nieuwe calibratiemethode?

C. Methodologie

Ik begin deze masterproef met een literatuurstudie, zodat ik een voldoende goed beeld heb van welke technieken er momenteel al toegepast worden, welke evaluatiecriteria er bestaan en welke de beste resultaten weergeven. Daarna ga ik op zoek naar een nieuwe calibratiemethode door verschillende zaken uit te testen. Eerst zal ik nagaan of de Brier score inderdaad beter wordt als men voor de verscheidene methoden hetzelfde aantal intervallen gebruikt. Dus de posities van de eindpunten van de intervallen zijn voor alle methoden dezelfde. Een andere techniek, die ik zal uitproberen, is het aantal cases in de intervallen constant te houden en vervolgens kijken hoe de Brier score hierbij evolueert. Bij deze techniek maak ik nog gebruik van twee varianten. De ene variant houdt rekening met de minimum refinement loss en bij de andere worden de drie gecalibreerde waarschijnlijkheden allemaal toegepast in de intervallen.

D. Inhoud van de thesis

Na deze inleiding bekijk ik in hoofdstuk twee welke evaluatiecriteria er allemaal bestaan, hoe deze berekend worden en waarvoor ze gebruikt kunnen worden. Bovendien wordt er op het begrip Brier score ingegaan.

In hoofdstuk drie worden de calibratietechniek, die toegepast wordt en het algoritme om gemeenschappelijke intervallen te verkrijgen, verduidelijkt.
Hoofdstuk vier bevat de uitleg omtrent de berekeningen die ik gemaakt heb, en de conclusies hierrond zijn terug te vinden in hoofdstuk vijf.
In hoofdstuk zes wordt de opbouw van de spreadsheets in MS Excel® uiteengezet. Met andere woorden, er wordt uitgelegd hoe alle calculaties met behulp van het spreadsheetprogramma tot stand zijn gekomen.
Tenslotte geeft het laatste hoofdstuk een overzicht van de belangrijkste conclusies. Daarnaast wordt er ook een aanbeveling gedaan voor verder onderzoek.

Hoofdstuk II : Evaluatiecriteria

Door de jaren heen zijn er een heleboel maatstaven ontstaan om classifiers te evalueren. Ferri, Hernández-Orallo en Modroiu (2009) en Caruana en Niculescu-Mizil (2004) bespreken een aantal van deze criteria en delen ze op in drie categorieën. In de eerste categorie zitten de evaluatiecriteria, die gebaseerd zijn op een threshold of een bepaalde waarde en deze zijn gebaseerd op een kwalitatief inzicht in de fout. De tweede groep zijn de maatstaven die gebaseerd zijn op een probabilistisch inzicht in de fout, dit wil zeggen dat de afwijking van de echte waarschijnlijkheid wordt berekend. Als laatste categorie hebben we deze criteria die gebaseerd zijn op hoe goed een model de voorbeelden rangschikt of classificeert.

Bij de bespreking van de volgende evaluatiecriteria wordt er gebruik gemaakt van de volgende notatie. Er is een testdataset gegeven waarbij m het aantal cases voorstelt en c het aantal klassen weergeeft. De werkelijke probabiliteit van observatie i om tot klasse j te behoren, wordt weergegeven door f(i,j). De auteurs veronderstellen dat f(i,j) altijd waarden aanneemt, die in het interval [0,1] liggen en het is geen waarschijnlijkheid, maar een indicatorfunctie. Het aantal observaties in klasse j wordt gedefinieerd door $m_j = \sum_{i=1}^{m} f(i,j)$. P(j) duidt de prior probabiliteit van de klasse j aan, $p(j) = m_j/m$. Als de classifier gegeven is, dan geeft p(i,j) de verwachte probabiliteit van observatie i om tot klasse j te behoren weer en neemt hij waarden aan in het interval [0,1]. C(i,j) is gelijk aan 1 als j de voorspelde klasse voor observatie i is, die verkregen wordt uit p(i,j) wanneer er gebruik gemaakt wordt van een bepaalde threshold. Anders is C(i,j) gelijk aan 0.

A. Threshold metrics

Deze maatstaven zijn gevoelig voor een goede keuze van de threshold. Tot deze categorie behoren de volgende evaluatiecriteria: accuracy, Kappa statistic, gemiddelde F-score, precision-recall break-even point, average precision, macro-averaged accuracy en de lift. Deze maatstaven worden gebruikt wanneer we willen dat een model het aantal fouten minimaliseert. Daardoor worden deze criteria veel gebruikt in directe toepassingen

van classifiers. In deze categorie zijn sommige maatstaven meer aangewezen voor evenwichtige of onevenwichtige datasets, voor signaal- of foutenopsporing of voor information retrieval, het opzoeken naar informatie in documenten.

Accuracy of nauwkeurigheid is de meest gebruikte maatstaf en is bovendien één van de eenvoudigste om een classifier te evalueren. Het wordt gedefinieerd als de mate van juiste voorspellingen van een model of omgekeerd, als het percentage van misclassificatiefouten. Het duidt het aandeel correcte voorspellingen aan, die de classifier maakt met betrekking tot de grootte van de dataset. Als een classifier continue output heeft, wordt er een drempel of threshold gekozen en alles boven deze drempel wordt als positief voorspeld . De formule voor de accuracy is de volgende:

$$Acc = \frac{\sum_{i=1}^{m}\sum_{j=1}^{c} f(i,j)C(i,j)}{m}$$

Oorspronkelijk is de **Kappa statistic** een criterium van overeenkomst tussen twee classifiers. Hoewel het ook gebruikt kan worden als een maatstaf om een classifier te beoordelen of om de gelijkenis tussen leden van een ensemble in multi-classifiers systemen te schatten. Het vormen van een ensemble gebeurt door het combineren van verschillende classifiers tot één grote classifier.

$$KapS = \frac{P(A) - P(E)}{1 - P(E)}$$

In deze formule is P(A) de relatieve waargenomen overeenkomst tussen twee classifiers en P(E) is de waarschijnlijkheid dat deze overeenkomst te wijten is aan toeval. In dit geval is P(A) gelijk aan de accuracy van de classifier, dus P(A) = Acc, die hierboven beschreven is. P(E) kan gedefinieerd worden als volgt:

$$P(E) = \frac{\sum_{k=1}^{c}([\sum_{j=1}^{c} f(i,k)C(i,j)] * [\sum_{j=1}^{c}\sum_{i=1}^{m} f(i,j)C(i,k)])}{m^2}$$

De **gemiddelde F-score** wordt vooral toegepast bij het zoeken naar informatie in documenten, naar documenten zelf, naar metadata die de documenten beschrijft en het zoeken binnen een aantal databases. De formule voor de F-score ziet er als volgt uit:

$$F - score\ (j) = \frac{2 * recall\ (j) * precision\ (j)}{recall\ (j) + precision\ (j)}$$

Recall kan omschreven worden als het aantal correct geclassificeerde positieven gedeeld door het totaal aantal positieven. Met andere woorden, dit is de fractie van de werkelijke positieven die ook als positief voorspeld zijn. **Precision** daarentegen is het aantal correct geclassificeerde positieven gedeeld door het totaal aantal dat als positief voorspeld wordt. Dit is de fractie van cases voorspeld als positief, die ook werkelijk positief zijn.

$$recall\ (j) = \sum_{i=1}^{m} \frac{f(i,j)C(i,j)}{m_j}$$

$$precision\ (j) = \frac{\sum_{i=1}^{m} f(i,j)C(i,j)}{\sum_{i=1}^{m_j} C(i,j)}$$

In de bovenstaande formules is j de index van de klasse die beschouwd wordt als "positief". Tot slot volgt dan de formule om de gemiddelde F-score te berekenen:

$$MFM = \frac{\sum_{j=1}^{C} F - Measure\ (j)}{c}$$

Precision-recall break-even point wordt gedefinieerd als de betrouwbaarheid op een bepaald punt (de threshold waarde) waar de precision en recall gelijk zijn. Een andere maatstaf die nog besproken wordt, is de **average precision**. Deze wordt berekend als het gemiddelde van betrouwbaarheden, die op elf gelijk uit elkaar geplaatste recallniveaus worden gezet.

Het rekenkundig gemiddelde van de gedeeltelijke nauwkeurigheden van elke klasse wordt de **macro average arithmetic (MAvA)** genoemd. Dit criterium staat soms ook wel bekend als het macrogemiddelde of macro average. De formule staat hieronder beschreven. Een andere maatstaf, die hier dicht bij aanleunt, is de **macro average geometric (MAvG)**. Dit wordt gedefinieerd als het geometrisch gemiddelde van de gedeeltelijke nauwkeurigheden van elke klasse.

$$MAvA = \frac{\sum_{j=1}^{c} \frac{\sum_{i=1}^{m} f(i,j)C(i,j)}{m_j}}{c}$$

$$MAvG = \sqrt[c]{\prod_{j=1}^{c} \frac{\sum_{i=1}^{m} f(i,j)C(i,j)}{m_j}}$$

Als laatste criterium is er de **lift**; deze wordt vaak gebruikt bij marketinganalyses. De lift meet hoeveel beter een classifier is in het voorspellen van positieven, dan een classifier die willekeurig positieven voorspelt. Meestal wordt de threshold zo bepaald dat er een vast percentage van de dataset als positief geclassificeerd wordt. De definitie van deze maatstaf is:

$$Lift = \frac{\%\ of\ true\ positives\ above\ the\ threshold}{\%\ of\ dataset\ above\ the\ threshold}$$

B. Probabiliteit metrics

De volgende maatstaven die besproken worden, zijn criteria die de afwijking tussen de geschatte probabiliteit met betrekking tot de werkelijke waarschijnlijkheid gaan kwantificeren. De gemiddelde absolute fout, de gemiddelde kwadratische fout, de LogLoss (cross-entropy), twee versies van waarschijnlijkheidsstandaarden en twee maatstaven voor calibratie maken deel uit van deze groep. Deze evaluatiecriteria zijn vooral nuttig wanneer er een beoordeling van de betrouwbaarheid van de classifiers nodig is. Er wordt niet alleen gemeten wanneer deze classifiers falen, maar ook of zij de verkeerde klasse met een hoge of een lage waarschijnlijkheid hebben geselecteerd. Dit is cruciaal bij machine ensembles om een gewogen fusie van de modellen behoorlijk uit te voeren.

De **mean absolute error** of de gemiddelde absolute fout toont hoeveel de voorspellingen van de echte probabiliteit afwijken. Deze maatstaf kan bepaald worden door middel van de onderstaande formule:

$$MAE = \frac{\sum_{j=1}^{c} \sum_{i=1}^{m} |f(i,j) - p(i,j)|}{m * c}$$

De volgende maatstaf is een kwadratische versie van de gemiddelde absolute fout die sterke afwijkingen van de echte waarschijnlijkheid bestraft. Dit criterium wordt de **gemiddelde kwadratische fout** genoemd en wordt nog verder in detail besproken in paragraaf D van dit hoofdstuk en neemt deze vorm aan:

$$MSE = \frac{\sum_{j=1}^{C}\sum_{i=1}^{m}(f(i,j) - p(i,j))^2}{m * c}$$

De **LogLoss** is ook een criterium dat weergeeft hoe goed de waarschijnlijkheidsschattingen zijn. Deze maatstaf wordt gebruikt als men geïnteresseerd is in het voorspellen van de waarschijnlijkheid dat een case positief is en het wordt voornamelijk gebruikt als calibratie belangrijk is. Om dit criterium te kunnen toepassen, heeft men deze formule nodig:

$$LogL = \frac{-\sum_{j=1}^{C}\sum_{i=1}^{m}(f(i,j)log_2 p(i,j))}{m}$$

Calibration loss wordt gedefinieerd als de gemiddelde kwadratische afwijking van de empirische waarschijnlijkheden die uit de helling van de ROC (Receiver operating characteristic) segmenten worden afgeleid.

$$CalLoss\ (j) = \sum_{b=1}^{r_j}\sum_{i \in S_{j,b}}\left(p(i,j) - \sum_{i \in S_{j,b}}\frac{f(i,j)}{|S_{j,b}|}\right)^2$$

In deze formule is r_j het aantal segmenten in de ROC curve voor klasse j, met andere woorden het aantal verschillende geschatte probabiliteiten voor klasse j:|{p(i,j)}|. Elk segment van de ROC curve wordt aangeduid door $s_{j,b}$ met b ∈ 1..r_j. Formeel wordt dit gedefineerd als $s_{j,b}$ = {i ∈ 1,…,m|∀ k ∈ 1,…,m : p(i,j) ≥ p(k,j) ∧ i ∉ $s_{j,d}$, ∀ d < b }. De algemene multiclass calibration loss wordt gedefinieerd als volgt:

$$CalL = \frac{1}{c}\sum_{j=1}^{c} CalLoss\ (j)$$

Een andere maatstaf is de **calibration by bins**, dit is een criterium dat gebaseerd is op overlapping van bins of segmenten. Voor elke klasse moeten alle observaties op voorspelde positieve klasse p(i,j) geordend worden en moeten er nieuwe indexen i* gegeven worden. De eerste 100 elementen, i* van 1 tot 100, behoren tot het eerste segment of bin. Daarna wordt het percentage van positieven (klasse j) in dit segment berekend als de werkelijke waarschijnlijkheid $\widehat{f}_{\text{j}}$. De fout van deze bin ziet er als volgt uit, $\sum_{i^* \in 1,\ldots,100} |p(i,j) - \widehat{f}_{\text{j}}|$. De tweede bin bevat dan de elementen van 2 tot 101 en de fout wordt dan op dezelfde manier berekend. Op het einde moet dan het gemiddelde van deze fouten genomen worden. Het probleem, als men gebruikt maakt van 100 elementen om een segment te vormen, is dat de bin te groot kan zijn voor kleine datasets. Daarom wordt de lengte van het segment bepaald door s = m/10, zo wordt het minder afhankelijk van de grootte van de dataset.

$$CAL\ (j) = \frac{1}{m-s} \sum_{b=1}^{m-s} \sum_{i^*=b}^{b+s-1} |p(i^*,j) - \frac{\sum_{i^*=b}^{b+s-1} f(i^*,j)}{s}|$$

Voor meer dan twee klassen wordt het criterium het gemiddelde van alle klassen, dit wordt als volgt genoteerd:

$$CalB = \frac{1}{c} \sum_{j=1}^{c} CAL(j)$$

Als laatste in deze categorie zijn er nog de **macro average mean probability rate (MAPR)** en de **mean probability rate (MPR)**. MAPR wordt berekend als een rekenkundig gemiddelde van de gemiddelde voorspellingen van elke klasse en heeft de volgende formule:

$$MAPR = \frac{\sum_{j=1}^{c} \frac{\sum_{i=1}^{m} f(i,j)p(i,j)}{m_{\text{j}}}}{c}$$

De mean probability rate is ook een maatstaf die de afwijking tot de werkelijke waarschijnlijkheid analyseert. Het is een niet-gelaagde versie van de MAPR, het rekenkundig gemiddelde van de geschatte probabiliteiten van de werkelijke klasse. MPR wordt berekend als volgt:

$$MPR = \frac{\sum_{j=1}^{C}\sum_{i=1}^{m} f(i,j)p(i,j)}{m}$$

C. Ordening/rangschikking metrics

De evaluatiecriteria, die van deze groep deel uitmaken, zijn criteria die de kwaliteit van het rangschikken kwantificeren. Deze maatstaven zijn belangrijk voor meerdere toepassingen, zoals customer relationship management, opsporing van fraude, filteren van spam, enz. Bij deze applicaties worden classifiers gebruikt om de beste n cases uit de dataset te selecteren of wanneer een goede scheiding van klassen cruciaal is.

De AUC (Area Under the ROC Curve of het gebied onder de ROC curve) van een binaire classifier is gelijkaardig aan de waarschijnlijkheid dat de classifier een willekeurig gekozen positieve case hoger rangschikt dan een willekeurig gekozen negatieve observatie.

$$AUC\ (j,k) = \frac{\sum_{i=1}^{m} f(i,j)\sum_{t=1}^{m} f(t,k)I(p(i,j),p(t,j))}{m_{\mathrm{j}} * m_{\mathrm{k}}}$$

I(.) is een vergelijkingsfunctie waarbij het volgende geldt:

- I(a,b) = 1 ⇔ a>b
- I(a,b) = 0 ⇔ a<b
- I(a,b) = 0,5 ⇔ a=b

Deze maatstaf is uitgebreid naar meerdere klassenproblemen in een aantal manieren. Hieronder worden daar vier varianten van voorgesteld.

Als eerste variant is er de **AUNU**, dit is de AUC van elke klasse ten opzichte van de rest, gebruikmakend van de uniforme klasseverdeling. Dit criterium berekent het gebied dat een c-dimensionele classifier behandelt als c twee-dimensionele classifiers heeft, waar klassen verondersteld worden een uniforme verdeling te hebben, zodat er een meting is, die onafhankelijk is van de verandering van de klasseverdeling. In deze formule verzamelt $\mathrm{rest_j}$ alle klassen verschillend van klasse j. Hier wordt het gebied onder de ROC curve berekend als het gemiddelde van de c combinaties.

$$AUNU = \frac{\sum_{j=1}^{C} AUC(j, rest_j)}{c}$$

AUNP is de tweede variant. Dit is de AUC van elke klasse ten opzichte van de andere klassen, gebruikmakend van de a priori klasseverdeling. Bij deze maatstaf wordt de c-dimensionele classifier als c twee-dimensionele classifier behandeld, waarbij er rekening gehouden wordt met de vorige probabiliteit van elke klasse (p(j)). De formule die bij dit criterium hoort, is de volgende:

$$AUNP = \sum_{j=1}^{c} p(j) AUC\ (j, rest_j)$$

De AUC van elke klasse tegen elkaar, gebruikmakend van de uniforme klasseverdeling is de **AU1U**. Deze maatstaf vertegenwoordigt de benadering van het gebied onder de ROC curve in het geval van multidimensionele classifiers. De AUC van c(c-1) binaire classifiers (alle mogelijke parencombinaties) wordt berekend en de uniforme verdeling van de klassen wordt in overweging genomen.

$$AU1U = \frac{1}{c(c-1)} \sum_{j=1}^{c} \sum_{k \neq j}^{c} AUC\ (j, k)$$

De laatste AUC is de AUC van elke klasse tegen elkaar, gebruikmakend van de a priori klasseverdeling (**AU1P**). Dit criterium vertegenwoordigt hetzelfde als de vorige beschreven AUC, maar houdt nu rekening met een a priori verdeling van de klassen.

$$AU1P = \frac{1}{c(c-1)} \sum_{j=1}^{c} \sum_{k \neq j}^{c} p(j)\, AUC\ (j, k)$$

De laatste maatstaven zijn nog twee evaluatiecriteria, die tussen de twee laatste categorieën inzitten, namelijk scored AUC (SAUC) en probabilistic AUC (PAUC). **Scored AUC** is een variant van AUC, waar de probabiliteiten in de definitie omvat zijn. Men wil een meting introduceren die robuust is om veranderingen met betrekking tot kleine

waarschijnlijkheidsvariaties te rangschikken. De onderstaande formule is de scored AUC voor twee klassen:

$$Scored\ AUC(j,k) = \frac{\sum_{j=1}^{m} f(i,j) \sum_{t=1}^{m} f(t,k) I\big(p(i,j), p(t,)\big) * (p(i,j) - p(t,k))}{m_{\mathrm{j}} * m_{\mathrm{k}}}$$

Dit is hetzelfde als AUC, met uitzondering van een extra term, die toegevoegd wordt om de afwijking van de waarschijnlijkheidsschatting te kwantificeren als de rangschikking niet juist is. De SAUC voor meerdere klassen wordt als volgt gedefinieerd:

$$SAUC = \frac{1}{c(c-1)} \sum_{j=1}^{c} \sum_{k \neq j}^{c} Scored\ AUC\ (j,k)$$

PAUC is ook een variant van de AUC, waar de probabiliteiten in de definitie zitten. Bij de **probabilistic AUC** wordt het deel met de indicatorfunctie niet meer gebruikt en dit betekent dat het geen juiste rankingsmaatstaf is. Voor twee klassen wordt de PAUC gedefinieerd als:

$$Prob\ AUC\ (j,k) = \frac{\sum_{i=1}^{m} \frac{f(i,j)p(i,j)}{m_{\mathrm{j}}} - \sum_{i=1}^{m} \frac{f(i,k)p(i,j)}{m_{\mathrm{k}}} + 1}{2}$$

Uit deze formule wordt dan de PAUC voor meerdere klassen bepaald en die ziet er dan als volgt uit:

$$PAUC = \frac{1}{c(c-1)} \sum_{j=1}^{c} \sum_{k \neq j}^{c} Prob\ AUC\ (j,k)$$

Cieslak en Chawla (2006) bespreken nog een ander evaluatiecriterium voor het beoordelen van de kwaliteit van waarschijnlijkheidsschattingen, dit is de 'negative cross entroyp' ($Loss_{NCE}$). Dit is het gemiddelde van 'negative cross entroyp' van het voorspellen van de echte waarde. Deze verliesmaatstaf meet hoe dichtbij de voorspelde waarden bij de werkelijke waarden van de klassen liggen.

D. Brier score

De situatie waarbij het controleschema de voorspeller op geen gunstige manier kan beïnvloeden, is gecreëerd door Brier (1950). Dit is het geval als de voorspellingen

uitgedrukt worden in probabiliteiten. Hij heeft een 'squared error loss statistic' voorgesteld dat voorspellende probabiliteiten met willekeurige resultaten vergelijkt. Deze score wordt de Brier score genoemd.

De Brier score wordt volgens Cieslak & Chawla (2006) als volgt gedefinieerd:

$$Loss_{\mathrm{BS}} = \frac{1}{n}\sum_{i=1}^{n}(y_{\mathrm{i}} - p_{\mathrm{i}})^2$$

Y_i is een aantal (n) onzekere gebeurtenissen waarbij $y_i = 1$ als de gebeurtenis plaatsvindt en $y_i = 0$ als de gebeurtenis niet plaatsvindt. P_i is de voorspellende probabiliteit van y_i.

De Brier score wordt als maatstaf ook wel herkend als de 'mean squared error' of gemiddelde kwadratische som (Ferri, Hernández-Orallo, & Modroiu, 2009 en Zadrozny & Elkan, 2001). Deze maatstaf bestraft de probabiliteiten die sterk afwijken van de echte waarschijnlijkheid.
Jakulin en Bratko (2003) definiëren 'error rate' of foutenkans als een speciaal geval van de Brier score voor deterministische classifiers. De Brier score kan bovendien een probabilitische classifier belonen voor het beter voorspellen of schatten van de waarschijnlijkheid.

De Brier score kan opgesplitst worden in calibration en refinement componenten. Calibration loss wordt als volgt gedefineerd (Cieslak & Chawla, 2006):

$$Loss_{\mathrm{C}} = \frac{1}{n}\sum_{j=1}^{k} n_{\mathrm{j}}\,(r_{\mathrm{j}} - p_{\mathrm{j}})^2$$

En de refinement loss wordt als volgt omschreven:

$$Loss_{\mathrm{R}} = \frac{1}{n}\sum_{j=1}^{k} n_{\mathrm{j}} r_{\mathrm{j}}\,(1 - r_{\mathrm{j}})$$

In deze formules is *k* het aantal gegenereerde unieke probabiliteiten, p_j gaat van j = 1 tot k voor de classifier op de testset en r_j is de fractie van observaties met klasse = 1 en die een waarschijnlijkheid toegewezen kregen van $p_j\left(r_j = \frac{y_j=1}{n_j}\right)$.

De Brier score opsplitsen in calibration loss en refinement loss is nuttig om de effecten van calibratie te begrijpen en om de gevolgen van het nemen van het gemiddelde te verstaan. Calibratie geeft een reeks van intervallen weer en een probabiliteit voor elk interval. Het eerstgenoemde heeft een effect op de refinement loss en het laatstgenoemde heeft een effect op de calibration loss.

Refinement loss neemt het gemiddelde van de wanorde in de waarschijnlijkheidstoewijzing over de verschillende intervallen. Het meet de stevigheid van de classifier. De refinement loss is het laagst als alle instanties een waarschijnlijkheid toegewezen krijgen, die dezelfde klasse met zich meebrengt. Als er een 50-50 splitsing tussen de klassetoewijzingen is, dan is de refinement loss het hoogst. Als een interval maar één element heeft, dan is de refinement loss nul. Als een classifier enkel unieke probabiliteiten heeft, is de refinement loss ook nul. Calibration loss weerspiegelt hoe goed het model de echte verspreiding van de data weergeeft (Cieslak & Chawla, 2006).

We werken volgend voorbeeld uit:
Stel dat we twee intervallen hebben met de volgende waarschijnlijkheden p_j, 0,5 en 0,55. Het eerste interval heeft 20 cases waarvan 10 voorbeelden als positief voorspeld zijn. Het andere interval heeft eveneens 10 positief voorspelde cases en een totaal van 20 observaties. We hebben dus een 50-50 splitsing in de klassentoewijzingen. De probabiliteiten r_j zijn voor beide hetzelfde en gelijk aan 0,5. Aangezien we het aantal positief voorspelde cases delen door het totaal aantal cases, dus 10/20. De berekening van de refinement loss ziet er dan als volgt uit:

$$Loss_r = \frac{1}{40}\left[\left(20 * 0{,}5 * (1 - 0{,}5)\right) + \left(20 * 0{,}5 * (1 - 0{,}5)\right)\right]$$

Het resultaat van deze calculatie is 0,250. We kunnen dus vaststellen dat de refinement loss hier optimaal is. Bovendien berekenen we calibration loss en deze calculatie wordt zo weergegeven:

$$Loss_c = \frac{1}{40}\left[20 * (0{,}5 - 0{,}5)^2 + 20 * (0{,}5 - 0{,}55)^2\right]$$

De uitkomst hiervan is gelijk aan 0,00125. Uit dit resultaat kunnen we afleiden dat de calibratie bijna 100% is, aangezien de calibration loss dicht bij nul ligt. Dit wil zeggen dat de classifier nauwkeurig voorspelt.

Naar de mening van Batra, Lenk en Wedel (2006) is calibratie gerelateerd aan vertekening en deze vertekening of bias is nul, wanneer de voorspellende probabiliteit en de relatieve frequenties gelijk zijn. Refinement daarentegen is gelijkaardig aan variantie en meet de tendens van het voorspellend model om waarden die dicht tegen één of nul liggen, voort te brengen, omdat in een goed gecalibreerd systeem de voorspellingen die dicht bij één of nul liggen, nuttiger zijn.

Flach en Matsubara (2007) definiëren calibration loss en refinement loss door gebruik te maken van de ROC curve. Calibration loss neemt het gemiddelde van de gekwadrateerde voorspellingsfout in elk segment van de ROC curve. De fout wordt genomen met betrekking tot de probabiliteit in een bepaald segment, wat het aantal positieven in het segment is en dus niet noodzakelijk nul of één is. Met andere woorden, calibration loss brengt de geschatte of voorspelde waarschijnlijkheden in verband met de empirische probabiliteiten die verkregen worden uit de segmenten van de ROC curve. De calibration loss is alleen nul als de ROC curve convex is. De tweede component, namelijk refinement loss is nul als en slechts als alle ROC segmenten horizontaal of verticaal zijn. De refinement loss houdt alleen rekening met de empirische waarschijnlijkheden, niet met voorspelde probabiliteiten. Daarom is het een maatstaf die niet enkel voor waarschijnlijkheidsschatters geëvalueerd kan worden, maar wel voor elke classifier die rangschikt.

Aangezien de Brier score de som is van de calibration loss en de refinement loss, is het moeilijk om beide te optimaliseren. Er bestaat een trade-off tussen deze twee elementen.

Waarschijnlijkheden r_j die dicht tegen nul of één liggen, zullen een lage refinement loss genereren, maar kunnen een hoge calibration loss produceren.
Het optimaliseren van de calibration loss is ideaal wanneer de waarschijnlijkheidsschatting in vergelijking met de empirische observatie nauwkeurig moet zijn. Refinement daarentegen is geschikt om een traditionele classificatietaak te optimaliseren. Nauwkeurige classifiers die met hoge waarschijnlijkheid voorspellen, zullen de beste refinementscores genereren.

De Brier score is een maatstaf voor het beoordelen van een prestatiemethode en wordt vooral gebruikt om de kwaliteit van schattingen van probabiliteiten te evalueren. Dit criterium wordt omschreven als het gemiddeld kwadratisch verlies dat voorkomt bij elke observatie in de testset. Het wijst die voorspellingen aan die de beste schattingen maken bij de werkelijke waarschijnlijkheden. Als de Brier score gebruikt wordt als evaluatiecriterium, dan moet men op zoek gaan naar de laagste Brier score. Hoe meer vertrouwen men heeft in het voorspellen van de daadwerkelijke klasse, hoe kleiner het verlies gaat zijn. Met andere woorden, hoe hoger de Brier score, hoe slechter de prestatie van de schatting (Jakulin & Bratko, 2003). Om de Brier score te minimaliseren, moet een voorspeller zowel goed gecalibreerd als verfijnd zijn.

Hoofdstuk III : Calibratietechniek en algoritme

A. Calibratietechniek

Er zijn twee soorten van classifiers: binaire classifiers en probabilistische classifiers. Bij binaire classifiers zijn er twee klassen, de ene klasse krijgt als label positief en de andere negatief. Volgens Vuk and Curk (2006) wordt 'precision' en 'accuracy' vaak gebruikt om de kwaliteit van de classificatie van binaire classifiers te meten. De probabilistische classifiers wijzen aan elk voorbeeld een score of waarschijnlijkheid toe die de juiste probabiliteit zou moeten uitdrukken dat een voorbeeld tot de positieve klasse behoort.

Niculescu-Mizil en Caruana (2007) bechrijven twee methoden om modelvoorspellingen in te delen in posterieure probabiliteiten: Platt calibratie en isotone regressie. Deze manieren kunnen enkel gebruikt worden voor binaire classificatie en het is niet onbelangrijk om deze uit te breiden voor problemen met meerdere klassen. Een manier om multiklasse problemen te behandelen, is om deze problemen om te zetten naar binaire problemen, de binaire modellen te calibreren en dan de voorspellingen opnieuw te combineren.

De Platt calibratie zet de SVM (support vector machine) voorspellingen om in posterieure waarschijnlijkheden door ze door een sigmoïde te laten passeren. Dit gebeurt door een transformatie via een sigmoïdefunctie, dit is een wiskundige functie en genereert een S-vormige curve. De output van het model waarop men leert, wordt voorgesteld door f(x). De ouput wordt dus door een sigmoïdefunctie omgevormd om gecalibreerde probabilteiten te verkrijgen.

$$P\left(y = 1 \middle| f\right) = \frac{1}{1 + \exp(Af + B)}$$

De parameters A en B worden aangepast door gebruik te maken van de maximum likelihood schatting van een fitting training set (f_i, y_i). Om de parameters A en B te vinden, wordt de 'gradient descent' toegepast, zodat zij de oplossing zijn voor:

$$argmin\, A, B \left\{ - \sum_i y_i \log(p_i) + (1 - y_i) \log(1 - p_i) \right\}$$

In de formule is $p_i = \frac{1}{1+\exp(Af_i+B)}$.

Bij deze methode moet men zich afvragen vanwaar de sigmoïde trainingsset komt. Als we dezelfde dataset om te calibreren gebruiken als deze waarop het model getraind is, dan treedt er ongewenste vertekening op. Bijvoorbeeld als het model leert om de trainingsset perfect te onderscheiden en het ordent daarbij alle negatieve voorbeelden voor de positieve observaties, dan zal de output van de sigmoïde transformatie enkel een 0,1 functie zijn. Dus er is een onafhankelijke calibratieset nodig om goede posterieure waarschijnlijkheden te verkrijgen.

Als men gebruik wil maken van probabilistische classifiers moet men, zo zeggen Fawcett en Niculescu-Mizil (2007), de voorspellingen van dit model eerst calibreren. Dit wil zeggen dat de scores van de observaties omgezet moeten worden in goed gecalibreerde posterieure probabiliteiten. De calibratietechniek die de auteurs hebben toegepast is het 'Pool Adjacent Violaters' (PAV) algoritme om een monotone stijgende transformatie van de scores van de classifier, die de laagste Brier score opbrengt, te vinden. Dit algoritme vindt een stapsgewijze constante oplossing voor het isotone regressieprobleem.

De calibratiemethode die gebruikt wordt, is gebaseerd op isotone regressie. Isotone regressie baseert zich op de veronderstelling dat de functie die de outputscores van de classifier in gecalibreerde posterieure waarschijnlijkheden omzet, monotoon stijgt met betrekking tot de score van de classifier. Isotone regressie deelt de waarschijnlijkheidsscores gegenereerd door het model in gecalibreerde probabiliteiten met een isotone (monotoon stijgende) mappingfunctie. Een hogere score wil zeggen dat er een hogere waarschijnlijkheid is dat de observatie positief is. Als het geleerde model f(x) gegeven is, dan is f_i de voorspelling van een model voor observatie x_i. Het ware label wordt gedefinieerd als y_i. De calibratie, die gebaseerd is op isotone regressie, neemt aan dat het model de cases correct rangschikt en daarom een niet-dalende mapping vormt, $y_i = m(f_i) + \varepsilon_i$. In deze formule is m de monotone stijgende functie. Isotone regressie als

zoektocht naar de monotone stijgende functie $\hat{m}$ met gegeven calibratieset (f_i,y_i) wordt omschreven als volgt:

$$\hat{m} = \arg min_z \sum (y_i - z\,(f_i))^2$$

Fawcett en Niculescu-Mizil (2007) leggen uit hoe het algoritme precies in zijn werk gaat. Een reeks van cases, gesorteerd door de scores die de classifier toegewezen heeft, is gegeven. Het PAV algoritme wijst eerst een probabiliteit van één toe aan elke positieve observatie en een waarschijnlijkheid van nul aan elke negatieve observatie. Daarna zet het algoritme elke instantie in zijn eigen groep. Bij elke iteratie gaat het algoritme bekijken welke de aangrenzende 'violaters' zijn. Dit zijn de aangrenzende groepen waarvan de probabiliteiten plaatselijk eerder stijgen dan dalen. Wanneer het zulke groepen vindt, gaat het deze groepen samenbrengen en worden hun waarschijnlijkheidsschattingen vervangen door het gemiddelde van de groepswaarden. Het algoritme gaat verder met dit proces van gemiddelden nemen en ze te vervangen totdat de hele reeks monotoon vermindert. Het resultaat is dan een serie van intervallen en een waarschijnlijkheidsschatting voor elk interval. Om een waarschijnlijkheidsschatting van een instantie te verkrijgen, zoeken we naar het interval waarvoor de observatie tussen de laagste en hoogste scores zich in het interval bevindt. Daarna wijzen we aan de instantie de waarschijnlijkheidsschatting van het interval toe. De pseudo-code van dit algortime is weergegeven in figuur 1.

Basic PAV method for generating probability estimates

Input: Scored training set (f_i , y_i), where f_i is the score assigned by the classifier and y_i is the correct class.

Output: Stepwise constant function generated by m

1: **begin**
2: Sort training set instances increasing by f_i
3: Put each training instance in its own group, $G_{i,i}$ and predict $m_{i,i} = y_i$
4: **while** $\exists G_{k,i-1}$ and $G_{i,l}$ ST $m_{k,i-1} \geq m_{i,l}$ **do**
5: Pool the instances in $G_{k,i-1}$ and $G_{i,l}$ into one group, $G_{k,l}$
6: $m_{k,l} = (\sum_{i=k}^{l} y_i) /(l - k + 1)$
7: Predict $m_{k,l}$ for all instances in $G_{k,l}$
8: **end while**
9: Output the stepwise constant function generated by m
10: **end**

Figuur 1 : Pseudo-code PAV algortime

De werking van het PAV algoritme kan verduidelijkt worden met een voorbeeld. Gegeven zijn vijftien cases, zes daarvan zijn negatief en negen zijn positief (zie fig. 2). Het PAV algoritme sorteert de cases in dalende volgorde rekening houdend met de score. Vervolgens wijst het algortime waarschijnlijkheidsschattingen toe aan de observaties. Elke positieve case krijgt een één en elke negatieve een nul. Het algortime zoekt herhaaldelijk naar aangrenzende 'violators' of overtreders: dit is een lokale, niet-monotoniciteit in de reeks. We zien in figuur 2 dat deze aangrenzende overtreders (een nul gevolgd door een één) aanwezig zijn bij volgende paren van cases 2-3, 6-7, 9-10 en 12-13.

#	Score	Probabilities						
		Initial	a1	a2	b	c1	c2	d
0	.9	1	1	1	1	1	1	1
1	.8	1	1	1	1	1	1	1
2	.7	0	0	0	0	0	0	3/4
3	.6	1	1	1	1	1	1	3/4
4	.55	1	1	1	1	1	1	3/4
5	.5	1	1	1	1	1	1	3/4
6	.45	0	0	0	0	1/2	2/3	2/3
7	.4	1	1	1	1	1/2	2/3	2/3
8	.35	1	1	1	1	1	2/3	2/3
9	.3	0	0	0	1/2	1/2	1/2	1/2
10	.27	1	1	1	1/2	1/2	1/2	1/2
11	.2	0	0	1/3	1/3	1/3	1/3	1/3
12	.18	0	1/2	1/3	1/3	1/3	1/3	1/3
13	.1	1	1/2	1/3	1/3	1/3	1/3	1/3
14	.02	0	0	0	0	0	0	0

Figuur 2: Voorbeeld van PAV algoritme

Het algoritme wordt van het einde van de reeks met cases naar het begin toe beschreven. In de eerste stap wordt de schending die door observaties 12 en 13 wordt gegenereerd, verwijderd. Dit gebeurt door het samenbrengen van deze twee cases en er wordt een waarschijnlijkheidsschatting van 1/2 toegewezen aan deze observaties (zie kolom a1 in figuur 2). Hierdoor onstaat er een nieuwe overtreding tussen case 11 en de aangrenzende groep 12-13. Om deze schending weg te werken, worden observatie 11 en de groep 12-13 samengenomen en wordt er een groep van drie instanties gevormd. De probabiliteitsschatting van deze groep wordt dan 1/3. Het resultaat bevindt zich in kolom a2 van figuur 2. Daarna worden cases 9 en 10 (één positief en één negatief) samengebundeld en er wordt een waarschijnlijkheidsschatting van 1/2 aan elke obseratie gegeven (zie kolom b in figuur 2).

In de volgende twee kolommen van figuur 2, c1 en c2 worden de overtredingen van observaties 6-8 (één negatief en twee positief) verwijderd in twee stappen. Eerst worden de cases 6 en 7 samengenomen en krijgen ze allebei een probabiliteitsraming van 1/2. Vervolgens worden observaties 6 tot 8 gebundeld en krijgen een waarschijnlijkheidsschatting van 2/3 toegewezen. De instanties 2-5 (één negatief en drie

positief) worden op een gelijkaardige manier in één groep samengebracht en er wordt een schatting aan gegeven van 3/4. Het uiteindelijke resultaat is weergegeven in kolom d van figuur 2. We kunnen vaststellen dat de reeks van waarschijnlijkheidsschattingen nu monotoon dalend is en dat er geen schendingen meer overblijven. Deze sequentie kan nu gebruikt worden als basis voor een functie die classifierscores omzet in waarschijnlijkheidsschattingen.

B. Algoritme voor gemeenschappelijke intervallen

Elke data-mining-techniek heeft een ander aantal intervallen als resultaat en de intervallen liggen ook niet op dezelfde plaats. We zouden een manier moeten vinden om de posities van de eindpunten van de intervallen voor elke methode constant te houden, zodat we de verschillende methoden zouden kunnen vergelijken.

Om dit te realiseren, kunnen we het algoritme dat in figuur 3 omschreven wordt, toepassen (Ludl & Widmer, 2000).

Given:

- A split point list s_1, s_2,…
- A merging parameter (minimal difference s)

The algorithm:

1. Sort the sequence of split points in ascending order.
2. Run through the sequence until you find two splits s_i and s_{i+1} with $s_{i+1} - s_i \leq s$.
3. Starting at $i + 1$ run through the sequence until you find two split points s_j and s_{j+1} with $s_{j+1} - s_j > s$.
4. Calculate the *median m* of $[s_i, \ldots, s_j]$.
   - If $s_j - s_i \leq s$ **merge** all split points in $[si, \ldots, sj]$ to m.
   - If $s_j - s_i > s$ **triple** the set of split points in $[si, \ldots, sj]$ to $[s_i, m, s_j]$.
5. Start at s_{j+1} and go back to step 2.

Figuur 3 : Algoritme om constante intervallen te verkrijgen

Vooraleer we dit kunnen gebruiken, hebben we eerst een lijst met split points nodig. Deze lijst gaan we creëren door de intervallen van alle methoden aan te wenden. De lijst met split points bestaat dus uit elk interval van elke methode en van deze intervallen wordt dan een reeks in stijgende volgorde gevormd. Voor we nu verder kunnen met het algoritme, moet er eerst een waarde voor de parameter s gekozen worden. Het algoritme zal toegepast worden met verschillende 'merging' parameters. De sequentie wordt vervolgens doorlopen, totdat er twee splits gevonden worden, waar het verschil tussen de tweede split (s_{i+1}) en de eerste split (s_i) kleiner of gelijk is aan de parameter s. Daarna wordt de reeks opnieuw doorlopen, totdat er twee splitpunten aangetroffen worden, waar het verschil tussen het tweede, nieuwe splitpunt (s_{j+1}) en het eerste, nieuwe splitpunt (s_j) groter is dan de 'merging' parameter. De berekening van de mediaan van het interval tussen s_i en s_j is de volgende stap van het algoritme. Als het verschil tussen s_j en s_i kleiner of gelijk is aan de gekozen parameter s, dan worden alle splitpunten in het interval samengenomen tot de mediaan van dit interval. Als het verschil kleiner is, dan wordt het interval herleid tot s_i, de mediaan m en s_j. Daarna gaan we verder met s_{j+1} en lopen we terug door het algoritme op zoek naar twee splits, waar het verschil kleiner of gelijk is aan s.

Als samenvatting kunnen we zeggen dat als de split points dichter bij elkaar liggen dan s, ze samengevoegd worden tot één punt, de mediaan. Als dit niet het geval is, dan wordt het gebied gekenmerkt door de mediaan en twee buitengrenzen.

Om het algoritme verder te verduidelijken, volgt hier een voorbeeld. Er zijn twee reeksen van intervallen, doordat er gebruik gemaakt is van twee verschillende classifiers. De eerste classifier heeft vier intervallen en dit zijn de volgende: 0,15; 0,30; 0,61 en 0,89. De tweede classifier heeft als resultaat een ander aantal intervallen, namelijk zes. De intervallen die bij de tweede classifier horen zijn deze: 0,10; 0,37; 0,52; 0,58; 0,64 en 1. We willen deze twee sequenties van intervallen nu samenvoegen en dat doen we door middel van het algoritme. Eerst moeten we een lijst met split points hebben en deze bekomen we door de intervallen in een stijgende volgorde te plaatsen. Deze lijst ziet er dan als volgt uit: 0,10; 0,15; 0,30; 0,37; 0,52; 0,58; 0,61; 0,64; 0,89 en 1. Met deze lijst van split points wordt dan het algoritme uitgevoerd.

Naast de lijst met split points moet er ook nog een merging parameter gedefinieerd worden. Deze merging parameter is het minimale verschil s. De eerste keuze die we gemaakt hebben, is om voor de merging parameter de waarde 0,03 te gebruiken. We gaan nu op zoek naar twee punten waarbij het verschil tussen deze twee punten kleiner is dan of gelijk aan de merging parameter. De eerste twee split points waarbij dit voorvalt, zijn het zesde punt (0,58) en het zevende split point (0,61). Het verschil tussen deze twee punten is net gelijk aan de merging parameter. Daarna zoeken we naar twee punten waarbij het verschil groter is dan 0,03. Dit vinden we bij punt acht (0,64) en punt negen (0,89). Hier is het verschil gelijk aan 0,25; wat duidelijk groter is dan 0,03. Nu we deze punten gevonden hebben, nemen we de mediaan van punt zes tot punt acht. De uitkomst van deze berekeningen is 0,61. Daarna trekken we split point acht (0,64) af van punt zes (0,58). Dit geeft als resultaat 0,06 en dit is meer dan de parameter s. Daarom wordt dit interval herleidt tot de twee eindpunten, 0,58 en 0,64, en de mediaan, die gelijk is aan 0,61. Als we nu verder de sequentie doorlopen, zien we dat er geen verschillen meer zijn die kleiner of gelijk zijn aan de parameter s. Er kan nu een nieuwe lijst van intervallen opgemaakt worden, rekening houdend met de uitvoering van het algoritme. Door deze intervallen in de juiste volgorde te plaatsen, merken we op dat deze intervallen dezelfde zijn als degene die we in het begin hadden. Dit wil dus zeggen dat de keuze van de parameter niet optimaal was. De berekeningen hiervan zijn terug te vinden in figuur 4.

s_{i+1}-s_i				s_{j+1}-s_j			
s_7-s_6=	0,03	<s	TRUE	s_9-s_8=	0,25	>s	TRUE
mediaan van $s_6 \rightarrow s_8$							
m =	0,61						
s_j-s_i							
s_8-s_6=	0,06	<s	FALSE				
		>s	TRUE				
→	si,m,sj						

Figuur 4 : Voorbeeld algoritme met s=0,03

Dezelfde lijst van intervallen wordt opnieuw gebruikt, maar nu met een andere keuze voor de merging parameter. We opteren deze keer voor 0,05. Punten één (0,10) en twee (0,15), zijn de eerste, waar het verschil kleiner is dan of gelijk aan de parameter s. Vervolgens zoeken we dus naar twee split points waar het verschil tussen deze twee groter is dan s en dit vinden we bij punt twee (0,15) en split point drie (0,30). Het verschil tussen deze punten is gelijk aan 0,15; wat opvallend groter is dan 0,05. Hierna maken we de calculatie van de mediaan tussen split point één en punt twee. De uitkomst van deze calculatie is gelijk aan 0,125. Nu trekken we punt één (0,10) af bij punt twee (0,15) en we bekomen als resultaat 0,05. Dit resultaat is gelijk aan de merging parameter en dus herleiden we dit interval tot de mediaan, die gelijk was aan 0,125 (zie figuur 5).

s_{i+1}-s_i				s_{j+1}-s_j			
s_2-s_1=	0,05	<s	TRUE	s_3-s_2=	0,15	>s	TRUE
mediaan van $s_1 \rightarrow s_2$							
m =	0,125						
s_j-s_i							
s_2-s_1=	0,05	<s	TRUE				
		>s	FALSE				
→	m						

Figuur 5 : Eerste stap voorbeeld algoritme met s=0,05

Daaropvolgend doorzoeken we verder de reeks van intervallen en vinden we bij punten zes (0,58) en zeven (0,61) weer een verschil dat kleiner is dan s. Bovendien is het verschil tussen split points acht (0,64) en negen (0,89) groter dan de merging parameter. We nemen nu de mediaan tussen de punten zes en acht en bekomen als resultaat het volgende, 0,61. Daarna wordt het verschil genomen tussen split point acht (0,64) en split point zes (0,58). Dit verschil is groter dan s, dus behouden we voor het interval allebei de eindpunten, 0,58 en 0,64, alsook de mediaan, 0,61 (zie figuur 6).

s_{i+1}-s_i				s_{j+1}-s_j			
s_7-s_6=	0,03	<s	TRUE	s_9-s_8=	0,25	>s	TRUE
mediaan van s_6 → s_8							
m =	0,61						
s_j-s_i							
s_8-s_6=	0,06	<s	FALSE				
		>s	TRUE				
→	si,m,sj						

Figuur 6 : Laatste stap voorbeeld algoritme met s=0,05

Verder zijn er geen twee punten meer waar het verschil kleiner is dan s, dus is het algoritme afgelopen. Nu kunnen we een nieuwe lijst met intervallen maken. Deze nieuwe reeks bestaat uit de volgende intervallen: 0,125; 0,30; 0,37; 0,52; 0,58; 0,61; 0,64; 0,89; 1. We kunnen vaststellen dat we nu nog 9 intervallen hebben in plaats van de 10, die we in het begin hadden. We kunnen eventueel deze intervallen nog herleiden naar een aantal dat kleiner is, door een andere merging parameter te kiezen. Met dit voorbeeld is de werking van het algoritme verduidelijkt en is het makkelijker te begrijpen.

Hoofdstuk IV : Berekeningen

Als eerste bestuderen we de dataset, welke variabelen er juist gebruikt worden. In het SPSS-bestand zien we dat er drie verschillende classifiers, namelijk discriminant-analyse (Disc), logistische regressie (Log) en een beslissingsboom (Tree) worden toegepast.

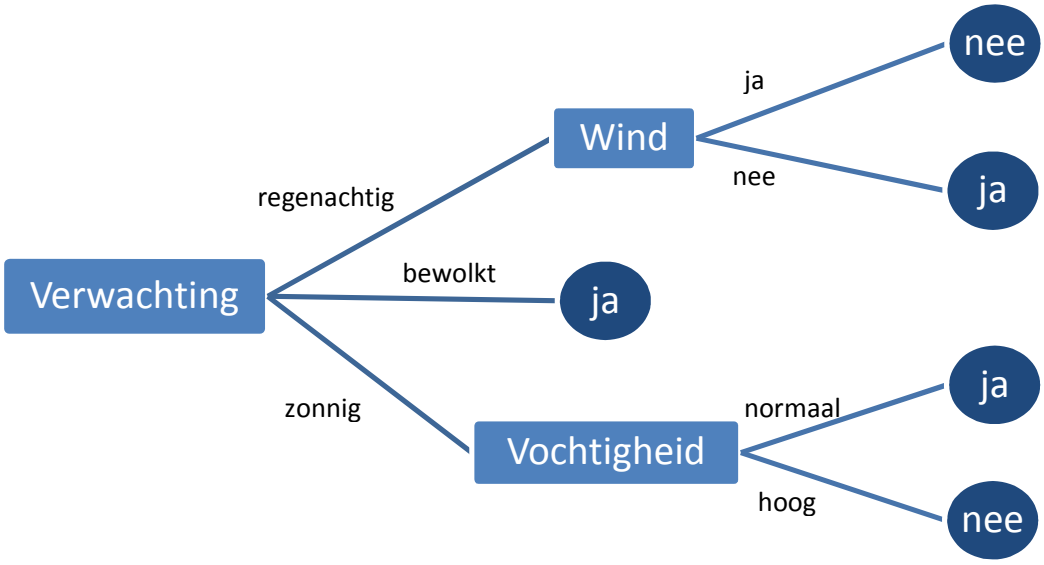
Discriminant-analyse wordt door Fernandez (2002) gedefinieerd als een multivariabele statistische techniek die in het algemeen toegepast wordt om een voorspellend, beschrijvend model te bouwen. Dit model, dat groepen onderscheidt, is gebaseerd op waargenomen voorspellende variabelen en classificeert elke observatie in één van de groepen. De verscheidene kwantitatieve attributen worden gebruikt om een enkele classificatievariabele te onderscheiden. Bij discriminant-analyse is vroegere kennis van de klassen noodzakelijk voor elke klasse. Deze informatie is meestal in de vorm van een steekproef. De algemene doelstellingen van discriminant-analyse worden als volgt omschreven :

- verschillen tussen groepen onderzoeken
- het effectief onderscheiden van groepen
- het identificeren van belangrijke discriminerende variabelen
- het toetsen van hypothesen op de verschillen tussen de verwachte groeperingen
- classificeren

Logistische regressie wordt gebruikt om de kans op een gebeurtenis te voorspellen. Baesens, Mues, en Vanthienen (2003) definiëren logistische regressie als een statistisch model dat de kans dat men tot groep 1 behoort, vormt als $P = (y = 1 \mid x_1, x_2, \ldots, x_p) = \frac{1}{(1+\exp(b_1x_1+b_2x_2+\cdots+b_px_p))}$. De ongekende parameters b_1, b_2,…,b_p worden geschat op basis van het trainingsvoorbeeld. De lineaire combinatie ($s = b_1x_1 + b_2x_2 + \ldots + b_px_p$) wordt berekend voor een nieuwe observatie. Op basis van deze score wordt dan de kans dat y = 1 bekomen.

Een **beslissingsboom** of 'decision tree' is een wetenschappelijk model voor de weergave van de alternatieven en keuzes in een besluitvormingsproces. Deze tool vergemakkelijkt

het kiezen tussen bepaalde acties. Het model wordt weergegeven in de vorm van een boom. De knooppunten in een beslissingsboom impliceren het testen van een bepaald attribuut. Gewoonlijk vergelijkt de test in een knooppunt de waarde van het attribuut met een constante. Een 'leaf node' geeft een classificatie, die op alle observaties die dit eindpunt bereiken van toepassing is. Om een nieuwe observatie te classificeren, wordt de beslissingsboom vanaf de top doorlopen. De waarden van de attributen worden getest in de opeenvolgende knooppunten. Wanneer een eindknooppunt bereikt is, is de observatie geclassificeerd en krijgt het de klasse van dit knooppunt toegewezen (Witten & Frank, 2000).



Figuur 7 : Voorbeeld van een beslissingsboom

In bovenstaande figuur is een voorbeeld weergegeven van een beslissingsboom. Met deze beslissingsboom zullen we nagaan of we tennis kunnen spelen of niet. Allereerst kijken we naar welke verwachting het weer heeft. Afhankelijk van de verwachting moeten we de vochtigheid meten en moeten we vaststellen of er wind is of niet. Stel dat de verwachting zonnig is, er is wind en de vochtigheid is normaal. Aangezien de verwachting zonnig is, kijken we hoe het met de vochtigheid gesteld is. De vochtigheid is normaal, dus komen we uit in het knooppunt ja. We kunnen ons partijtje tennis spelen. Dus afhankelijk van de

parameters die met de attributen worden meegegeven, krijgen we een ander resultaat uit onze beslissingsboom. Met dit hulpmiddel is het makkelijk om een keuze te maken als de waarden van een aantal attributen op voorhand gegeven zijn.

Voor het berekenen van de Brier score wordt er gebruik gemaakt van MS Excel®. MS Excel® is een spreadsheetprogramma, dat over de hele wereld gebruikt wordt, waarmee je gegevens kan berekenen, ordenen, vergelijken en presenteren. Met dit programma is het makkelijk om een goed overzicht te geven van de noodzakelijke calculaties. Een ander voordeel van een spreadsheet is, dat het eenvoudig is om data aan te passen en te updaten. Bovendien kan men mathematische formules toepassen om resultaten te genereren.

Om het resultaat beter en geloofwaardiger te maken, worden de berekeningen niet uitgevoerd op slechts één bestand. Dezelfde calculaties worden gedaan op 9 bestanden met allemaal verschillende data. Hierbij maken we gebruik van een 'cross-validation setting'. Bij 'cross-validation' wordt de data in een aantal groepen van dezelfde grootte onderverdeeld.

Uit de dataset kunnen we vaststellen dat er van elke classifier twee variabelen gecreëerd zijn. De ene variabele geeft de gegevens weer van de classifier zonder dat er calibratie toegepast is, dit zijn de variabelen die het voorvoegsel 'Prob' hebben. Als de variabele begint met 'Calibrated' wil dit zeggen dat deze variabele de observaties bevat waarop een calibratietechniek is toegepast. Het totale aantal observaties is voor elke techniek en variabele gelijk, namelijk 610.

Voor we met de berekeningen kunnen starten, maken we eerst met behulp van de tool 'Case summaries' in SPSS een overzicht van een bepaalde variabele. We gaan de observaties van de verschillende methoden groeperen naar probabiliteit. Met deze tool verkrijgen we voor elke waarschijnlijkheid het aantal observaties dat bij deze probabiliteit hoort (= kolom N). Bovendien vragen we de gegevens op waarbij het ware label y = 1 is; hiervan wordt dan de som gemaakt (= kolom Sum). Voor de beslissingsboom van het

eerste databestand staat de output van deze tool in figuur 8, zowel voor de gegevens waarop calibratie is toegepast, als deze waarop er geen calibratietechniek is toegepast.

Case Summaries

Prob_Tree	N	Sum
,147	162	25
,242	38	9
,336	83	24
,350	30	13
,710	29	20
,750	34	22
,788	22	20
,818	208	168
1,000	4	4
Total	610	305

Case Summaries

Calibrated_Tree	N	Sum
,170731707317	162	25
,286274509804	121	33
,347826086957	30	13
,663716814159	63	42
,674418604651	22	20
,840277777778	212	172
Total	610	305

Figuur 8 : Case summaries beslissingsboom databestand 1

In de dataset is er een twee-klassen probleem, y ∈ {0,1}, waar 1 de positieve of meerderheidsklasse is en 0 de negatieve of minderheidsklasse. De waarschijnlijkheid dat de positieve klasse voorspeld wordt voor observatie i, is p_i. Als we naar de formule kijken om de Brier score te bepalen, $Loss_{\mathrm{BS}} = \frac{1}{n}\sum_{i=1}^{n}(y_i - p_i)^2$, dan zien we dat de kwadratische term sommeert over alle mogelijke waarschijnlijkheidswaarden die worden toegewezen aan de testinstantie, wat in dit geval twee is. Het slechste geval zou zijn wanneer $p_i = 0$ en het ware label y = 1. Dit zou leiden tot $(1-0)^2 + (0-1)^2 = 2$. Het beste geval daarentegen zou zijn wanneer $p_i = 1$ en het echte label y = 1. Dit zou dan leiden tot het volgende, $(1-1)^2 + (0+0)^2 = 0$.

A. Methode met algoritme gemeenschappelijke intervallen

Door middel van de tool case summaries wordt de Brier score voor alle methoden (beslissingsboom, logistische regressie en discriminant-analyse) berekend. Zoals hierboven beschreven, kan de Brier score worden opgesplitst in calibration loss en refinement loss. Om een resultaat voor deze score te krijgen, werd dus eerst de calibration en refinement loss apart berekend en daarna worden deze twee resultaten opgeteld om de Brier score te bekomen.

Om te beginnen wordt de calibration loss, $Loss_{\mathrm{C}} = \frac{1}{n}\sum_{j=1}^{k} n_{\mathrm{j}}\,(r_{\mathrm{j}} - p_{\mathrm{j}})^2$, berekend. Voor we alles in de formule kunnen invullen, moeten we eerst de r_j bepalen. Deze wordt gevonden door de som van de gegevens, waarbij het ware label, y = 1 te delen is door het aantal gegevens dat bij deze probabiliteit hoort, n_j. Daarna is de formule in deeltjes opgedeeld om het berekenen te vergemakkelijken. Hierna wordt dan het kwadraat van het verschil tussen de r_j en de p_j genomen. Vervolgens wordt dit kwadraat vermenigvuldigd met n_j. Van deze resultaten wordt de som genomen en gedeeld door het aantal gegevens in de dataset, n = 610. Zo bekomen we de calibration loss (zie figuur 9).

Case Summaries

V9

Prob_Tree	N	Sum	rj	(rj-pj)²	nj*(rj-pj)²
0,147	162	25	0,154320988	5,35969E-05	0,008682691
0,242	38	9	0,236842105	2,66039E-05	0,001010947
0,336	83	24	0,289156627	0,002194302	0,182127036
0,350	30	13	0,433333333	0,006944444	0,208333333
0,710	29	20	0,689655172	0,000413912	0,012003448
0,750	34	22	0,647058824	0,010596886	0,360294118
0,788	22	20	0,909090909	0,014663008	0,322586182
0,818	208	168	0,807692308	0,000106249	0,022099692
1,000	4	4	1	0	0
Total	610	305			

Som nj*(rj-pj)²	Loss c
1,117137448	0,002

Figuur 9 : Calculatie calibration loss voor beslissingsboom van databestand 1

Als tweede wordt de calculatie gemaakt van de refinement loss, $Loss_{\mathrm{R}} = \frac{1}{n}\sum_{j=1}^{k} n_{\mathrm{j}} r_{\mathrm{j}}\,(1 - r_{\mathrm{j}})$. Ook deze berekening is opgesplitst in verschillende stukjes. Hier vermenigvuldigen we eerst n_j en r_j met elkaar. Dan wordt het verschil genomen tussen één en r_j en de volgende stap is de vermenigvuldiging van de twee laatste termen. Hierna worden deze uitkomsten gesommeerd, gedeeld door n en zo wordt de refinement loss verkregen (zie figuur 10).

Case Summaries

V9

Prob_Tree	N	Sum	nj*rj	1-rj	(nj*rj)*(1-rj)
0,147	162	25	25	0,845679012	21,14197531
0,242	38	9	9	0,763157895	6,868421053
0,336	83	24	24	0,710843373	17,06024096
0,350	30	13	13	0,566666667	7,366666667
0,710	29	20	20	0,310344828	6,206896552
0,750	34	22	22	0,352941176	7,764705882
0,788	22	20	20	0,090909091	1,818181818
0,818	208	168	168	0,192307692	32,30769231
1,000	4	4	4	0	0
Total	610	305			

Som (nj*rj)*(1-rj)	Loss r
100,5347806	0,165

Figuur 10 : Calculatie refinement loss voor beslissingsboom van databestand 1

Hierna berekenen we de Brier score voor de beslissingsboom van de gegevens waarop geen calibratie toegepast is. Dit is de optelling van de calibration en refinement loss. Voor databestand 1 is het resultaat dan 0,167.

De Brier score wordt voor alle drie de methoden berekend, dus zowel voor de beslissingsboom, de logistische regressie als de discriminant-analyse. Tabel 1 geeft het overzicht weer van de resultaten van de Brier score voor de gegevens waarop geen calibratie toegepast is voor databestand 1.

Tabel 1 : Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	**Disc**	**Tree**
Brier score	**0,185**	**0,207**	**0,167**
Calibration loss	/	/	**0,002**
Refinement loss	/	/	**0,165**

De logistische regressie en de discriminant-analyse hebben gedetailleerde schattingen, het is daardoor dat de refinement loss nul is en de Brier score dus gelijk is aan de calibration loss. De Brier score van de beslissingsboom daarentegen bestaat uit een refinement loss en een calibration loss. Dit komt omdat de beslissingsboom voor elk interval een schatting genereert.

In tabel 2 staan de resultaten van de Brier score van de gegevens waarop wel een calibratietechniek toegepast is. Hier kunnen we vaststellen dat de logistische regressie en de discriminant-analyse wel een calibration en refinement loss hebben. Er zijn namelijk intervallen gegenereerd en voor elk interval is er een schatter gegeven door de calibratiemethode.

Tabel 2 : Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,182**	**0,193**	**0,168**
Calibration loss	**0,005**	**0,013**	**0,003**
Refinement loss	**0,177**	**0,180**	**0,165**

Na de berekening van de Brier score op de gegevens waarop geen calibratie toegepast is, wordt het algoritme om gemeenschappelijke intervallen te krijgen, uitgevoerd. Zoals eerder beschreven staat, moet er eerst een lijst van split points zijn, vooraleer het algoritme kan worden toegepast. Om deze lijst op te maken, worden de intervallen die door de calibratietechniek gecreëerd zijn, gebruikt. Deze intervallen zijn voor databestand 1 terug te vinden in figuur 11.

Calibrated_Tree
0,170731707
0,286274510
0,347826087
0,663716814
0,674418605
0,840277778

Calibrated_Log
0
0,126262626
0,197530864
0,227272727
0,281553398
0,286458333
0,500000000
0,535714286
0,666666667
0,681415929
0,684210526
0,732142857
0,811594203
0,821917808
0,868421053
0,925925926
1

Calibrated_Disc
0
0,076923077
0,100000000
0,121212121
0,125000000
0,177570093
0,234042553
0,241935484
0,275229358
0,285714286
0,300000000
0,370967742
0,380000000
0,400000000
0,461538462
0,567164179
0,627906977
0,641975309
0,744444444
0,750000000
0,785714286
0,806451613
0,846153846
0,888888889
0,923076923
0,928571429
1

Figuur 11 : Intervallen gecreëerd door calibratietechniek voor databestand 1

De intervallen van alle drie de methoden worden in een stijgende volgorde geplaatst, te beginnen bij nul, en dit vormt dan de lijst met split points (zie figuur 12). Voor databestand 1 wordt er dan een reeks opgesteld die bestaat uit 48 split points.

	Split points
s_1	0
s_2	0,076923077
s_3	0,100000000
s_4	0,121212121
s_5	0,125000000
s_6	0,126262626
s_7	0,170731707
s_8	0,177570093
s_9	0,197530864
s_{10}	0,227272727
s_{11}	0,234042553
s_{12}	0,241935484

	Split points
s_{13}	0,275229358
s_{14}	0,281553398
s_{15}	0,285714286
s_{16}	0,286274510
s_{17}	0,286458333
s_{18}	0,300000000
s_{19}	0,347826087
s_{20}	0,370967742
s_{21}	0,380000000
s_{22}	0,400000000
s_{23}	0,461538462
s_{24}	0,500000000

	Split points
s_{25}	0,535714286
s_{26}	0,567164179
s_{27}	0,627906977
s_{28}	0,641975309
s_{29}	0,663716814
s_{30}	0,666666667
s_{31}	0,674418605
s_{32}	0,681415929
s_{33}	0,684210526
s_{34}	0,732142857
s_{35}	0,744444444
s_{36}	0,750000000

	Split points
s_{37}	0,785714286
s_{38}	0,806451613
s_{39}	0,811594203
s_{40}	0,821917808
s_{41}	0,840277778
s_{42}	0,846153846
s_{43}	0,868421053
s_{44}	0,888888889
s_{45}	0,923076923
s_{46}	0,925925926
s_{47}	0,928571429
s_{48}	1

Figuur 12 : Split points voor databestand 1

Nu deze reeks van split points gekend is, moet de parameter *s* nog gedefinieerd worden. Voor databestand 1 zijn er een aantal mergingparameters uitgetest. De bedoeling is om zo weining mogelijk intervallen over te houden. Als eerste keuze wordt er geopteerd om s gelijk te stellen aan 0,01. Maar met deze keuze worden er nog altijd veel intervallen bekomen, namelijk vijfendertig. Daarna wordt s gelijkgesteld aan 0,05. Dit blijkt een goede parameter te zijn, aangezien er nog elf intervallen overblijven.

De optimale merging parameter s blijkt voor de meeste databestanden gelijk te zijn aan 0,05 of 0,06. Er zijn maar een paar uitzonderingen waarbij deze waarden geen goed resultaat geven.Dit wil zeggen waar er nog teveel intervallen resteren.

s_{i+1}-s_i				s_{j+1}-s_j			
s_3-s_2=	0,023076923	<s	TRUE	s_{23}-s_{22}=	0,061538462	>s	TRUE
mediaan van $s_2 \rightarrow s_{22}$							
m =	0,241935484						
s_j-s_i							
s_{22}-s_2=	0,323076923	<s	FALSE				
		>s	TRUE				
→	si,m,sj						

Figuur 13 : Stap 1 van algoritme gemeenschappelijke intervallen voor databestand 1

Nu s vastligt op een bepaalde waarde kan het algoritme toegepast worden. Als eerste stap wordt er op zoek gegaan naar twee opeenvolgende split points waar het verschil tussen deze twee kleiner is dan de parameter s. Bij split point 2 en 3 zien we dat dit verschil inderdaad kleiner is dan de waarde van s, namelijk 0,023076923. Daarna wordt er een verschil gezocht tussen twee opeenvolgende split points waarbij dit verschil groter is dan 0,05. Dit komt voor bij split point 22 en 23 en dit heeft de waarde van 0,061538462. Vervolgens wordt de mediaan berekend van het interval tussen split 2 en 22 en het resultaat is gelijk aan 0,241935484. Als laatste stap wordt het verschil genomen tussen split point 22 en split point 2. In dit geval is de uitkomst groter dan de merging parameter en worden het eerste split point (2), de mediaan (0,241935484) en het laatste split point (22) als waarden voor de intervallen genomen (zie figuur 13). Zo wordt de hele reeks doorlopen (zie figuur 14) en wordt er als resultaat een nieuwe lijst met intervallen verkregen.

s_{i+1}-s_i				s_{j+1}-s_j			
s_{24}-s_{23}=	0,038461538	<s	TRUE	s_{27}-s_{26}=	0,060742798	>s	TRUE
mediaan van s_{23} → s_{26}							
m =	0,517857143						
s_j-s_i							
s_{26}-s_{23}=	0,105625718	<s	FALSE				
		>s	TRUE				
→	si,m,sj						

s_{i+1}-s_i				s_{j+1}-s_j			
s_{28}-s_{27}=	0,014068332	<s	TRUE	s_{48}-s_{47}=	0,071428571	>s	TRUE
mediaan van s_{27} → s_{47}							
m =	0,785714286						
s_j-s_i							
s_{47}-s_{27}=	0,300664452	<s	FALSE				
		>s	TRUE				
→	si,m,sj						

Figuur 14 : Stap 2 en 3 van algoritme gemeenschappelijke intervallen voor bestand 1

De gemeenschappelijke intervallen die dan voor databestand 1 verkregen worden, worden weergegeven in figuur 15. Deze worden hierna gebruikt om opnieuw de Brier score voor de verschillende methoden te berekenen.

	Nieuwe unieke probabiliteiten
1	0
2	0,076923077
3	0,241935484
4	0,400000000
5	0,461538462
6	0,517857143
7	0,567164179
8	0,627906977
9	0,785714286
10	0,928571429
11	1,000000000

Figuur 15 : Verkregen intervallen na toepassing algoritme voor databestand 1

Nadat deze intervallen gegenereerd worden, kunnen we opnieuw de Brier score voor de verschillende methoden berekenen, maar nu met de gemeenschappelijke intervallen. Deze calculatie gebeurt op dezelfde manier als hierboven beschreven werd. De gemeenschappelijke intervallen worden zowel gebruikt bij de gegevens waarop geen calibratietechniek toegepast is als bij de gegevens waarbij wel calibratie gebruikt is. De resultaten van deze berekening zijn te vinden in tabel 3 en 4.

Tabel 3 : Brier score met 11 gemeenschappelijke intervallen, toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,190**	**0,210**	**0,176**
Calibration loss	**0,011**	**0,018**	**0,010**
Refinement loss	**0,179**	**0,192**	**0,166**

Tabel 4 : Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,190**	**0,200**	**0,178**
Calibration loss	**0,009**	**0,007**	**0,010**
Refinement loss	**0,181**	**0,193**	**0,168**

Uit deze tabellen kunnen we concluderen dat de beslissingsboom als beste model naar voren komt. Dit zowel bij de gecalibreerde gegevens als bij de gegevens waarop geen calibratietechniek toegepast is. Vooral bij de discriminant-analyse heeft de calibratie een gunstig resultaat, wat wil zeggen dat de Brier score lager is. Dit komt vooral door de verlaging van de calibration loss, namelijk met 0,011. Voor de logistische regressie blijft de score gelijk, maar is er een verschuiving van de calibration loss naar de refinement loss van 0,002. De decision tree is dan wel het beste model volgens dit evaluatiecriterium, maar we zien wel dat deze score met calibratie omhoog gaat met 0,002. Deze stijging is te wijten aan de refinement loss, omdat de calibration loss dezelfde blijft. De resultaten van de berekeningen van de andere bestanden zijn te consulteren in bijlage 1.

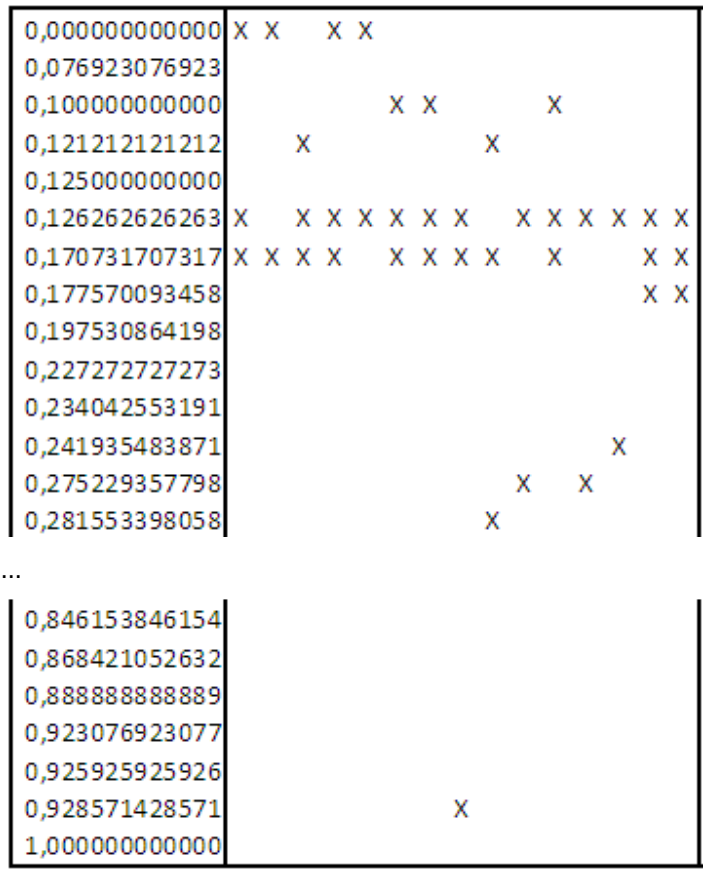
B. Methode met minimum aantal cases in interval

1. Gebruikmakend van minimum refinement loss

Bij de berekeningen van de vorige methode is er een fenomeen dat opvalt. We kunnen vaststellen dat er een probleem is bij de meeste calculaties. We kunnen duidelijk zien dat er bij vele intervallen geen cases voorkomen. Dit zorgt voor moeilijkheden bij het berekenen van de Brier score, omdat we dan gaan delen door nul. R_j wordt gedefinieerd als de som van de cases waarbij y = 1 gedeeld wordt door het totaal aantal cases. Door deze definitie krijgen we, als er geen cases in een interval zitten, een nul in de noemer. Omwille van dit probleem proberen we een andere manier uit om gemeenschappelijke intervallen te verkrijgen voor de drie methoden.

Aangezien elke dataset uit drie verschillende methoden bestaat, hebben we voor elk bestand dan ook drie opdelingen in intervallen, voor elke methode één opdeling. Deze indeling in intervallen komt voort uit de calibratietechniek. Deze techniek zorgt voor het genereren van deze intervallen. Vervolgens worden de intervallen van de beslissingsboom, de logistische regressie en de discriminant-analyse samengenomen. Ze worden in een stijgende volgorde, beginnend bij nul, geplaatst. Deze volgorde is dezelfde als de lijst met split points, die voor de eerste methode gebruikt werd.

Één enkele case heeft drie gecalibreerde probabiliteiten, één voor elke methode. Deze gegevens worden uit het databestand gehaald. Dit zijn de waarschijnlijkheden per methode die we bij de variabele met voorvoegsel 'Calibrated' terugvinden. Nu moet er per case bepaald worden bij welke interval deze case thuishoort. De drie gecalibreerde waarschijnlijkheden worden in de lijst van intervallen geplaatst. Dit gebeurt door middel van een functie in MS Excel®. Om te zien in welk interval een probabiliteit ligt, kijken we of de probabiliteit groter of gelijk is aan het eerste eindpunt en of het kleiner is dan het tweede eindpunt. Als dit het geval is, dan wordt er een X geplaatst in het vakje en behoort deze waarschijnlijkheid tot dit interval. Zo gebeurt dit voor de drie gecalibreerde probabiliteiten en voor alle cases. In figuur 16 zie je een deel van het resultaat voor de eerste vijftien cases van het eerste databestand.



Figuur 16 : Plaatsing eerste 15 cases van databestand 1 in interval

Nadat alle gecalibreerde probabiliteiten in een interval geplaatst zijn, moeten we nagaan of ze allemaal in een verschillend interval zitten of niet. Dit wordt andermaal vastgesteld door middel van een MS Excel® functie, namelijk 'COUNTIF'. Deze functie telt het aantal cellen op die binnnen een vastgesteld bereik voldoen aan een bepaalde conditie. Er moeten dus twee parameters opgegeven worden, namelijk het bereik en het criterium waaraan de cel moet voldoen. We tellen nu per case het aantal cellen op waar er een X in de cel staat. Het resultaat van deze formule moet minimum één zijn, de drie waarschijnlijkheden behoren tot eenzelfde interval en mag maximum drie zijn, de drie probabiliteiten liggen allemaal in een verschillend interval.

Als de probabiliteiten in drie verschillende intervallen liggen, dan moeten we gaan bepalen voor welk interval we deze case gaan meetellen. De minimum refinement loss

wordt hiervoor als criterium gebruikt. We berekenen de refinement loss voor de waarschijnlijkheden door middel van de volgende formule, p*(1-p), waar p voor de probabiliteit staat. Daarna gaan we op zoek naar de waarschijnlijkheid met de laagste refinement loss via de MS Excel® functie 'MIN'. Deze functie geeft het laagste cijfer uit een reeks van waarden weer. Het bijbehorende interval wordt gebruikt om het aantal overgebleven intervallen te bepalen. In het geval dat er maar twee intervallen zijn, maken we gebruik van de meerderheidsregel of 'majority rule' om het interval te bepalen. Het interval waarin er twee probabiliteiten vallen, wordt als bepalend beschouwd en dit interval houden we voor het vastleggen van het aantal intervallen.

Een voorbeeld volgt om het geheel toe te lichten. Een case heeft de volgende drie gecalibreerde probabiliteiten: 0,076923, 0,286458 en 0,170731. Deze drie waarschijnlijkheden worden allemaal in een verschillend interval geplaatst en wel als volgt:

0,076923 → [0,076923; 0,1]
0,286458 → [0,286458;0,3]
0,170731 → [0,170731; 0,177570]

Nu we hebben kunnen vaststellen dat deze drie waarden in een verschillend interval liggen, gaan we op zoek naar de minimum refinement loss. Zoals hierboven beschreven, wordt de refinement loss berekend door p*(1-p). In ons voorbeeld geeft dit het volgende:

0,076923*(1-0,076923) = 0,071006
0,286458*(1-0,286458) = 0,204399
0,170731*(1-0,170731) = 0,141582

Uit deze reeks van waarden halen we het getal met de laagste refinement loss. In dit geval is dit 0,071006. Deze case ligt in het interval [0,076923; 0,1] en wordt dus verder gebruikt bij het bepalen van de gemeenschappelijke intervallen.

Vervolgens bekomen we een reeks van waarden die we gaan gebruiken om het aantal intervallen te reduceren. Deze waarden worden met behulp van SPSS in de lijst met intervallen, die we bekomen hebben door de intervallen van de drie methoden samen te nemen, geplaatst. Eerst worden de waarden in verschillende variabelen omgezet. Dit wil zeggen dat bijvoorbeeld het interval van 0 tot 0,07, de waarde 0 krijgt. Het interval tussen 0,07 en 0,1 krijgt dan de waarde 0,07 mee, enz. Daarna wordt er door SPSS een tabel opgemaakt met het aantal cases in elk interval. Het resultaat voor databestand 1 is terug te vinden in figuur 17.

		Frequency	Percent
Valid	,000000000000	6	1,0
	,076923076923	4	,7
	,100000000000	12	2,0
	,121212121212	13	2,1
	,125000000000	6	1,0
	,126262626263	67	11,0
	,170731707317	100	16,4
	,177570093458	4	,7
	,197530864198	11	1,8
	,227272727273	2	,3
	,234042553191	1	,2
	,241935483871	4	,7
	,275229357798	12	2,0
	,281553398058	10	1,6
	,285714285714	2	,3
	,286274509804	44	7,2
	,286458333333	10	1,6
	,347826086957	7	1,1

	Frequency	Percent
,663716814159	5	,8
,674418604651	1	,2
,681415929204	14	2,3
,684210526316	2	,3
,732142857143	5	,8
,744444444444	7	1,1
,750000000000	1	,2
,785714285714	2	,3
,811594202899	24	3,9
,821917808219	6	1,0
,840277777778	121	19,8
,846153846154	5	,0
,868421052632	18	3,0
,888888888889	14	2,3
,923076923077	5	,8
,925925925926	35	5,7
,928571428571	15	2,5
1,000000000000	15	2,5
Total	610	100,0

Figuur 17 : Aantal cases in elk interval voor databestand 1

Uit deze figuur kunnen we vaststellen dat de intervallen waar er geen cases inzitten, dus het aantal nul is, zijn weggelaten. Voor databestand 1 houden we dan 36 intervallen over. Aangezien dit nog veel is, moeten we proberen om deze intervallen nog te reduceren. Een manier waarop we dit zouden kunnen doen, is een regel opleggen dat een interval een minimum aantal cases moet bevatten om opgenomen te mogen worden. De eerste norm die opgelegd wordt, is dat er minimaal 10 cases in een interval moeten zitten, opdat het

interval zou mogen blijven bestaan. Als we deze regel toepassen voor databestand 1, zien we dat er nu nog 17 intervallen overblijven. Het volgende dat we proberen, is het minimum aantal cases in één interval vast te leggen op 15. Met deze norm houden we nog 9 intervallen over (zie figuur 18).

Minimum 10 cases in 1 interval	
0,100000000000	22
0,121212121212	13
0,126262626263	73
0,170731707317	100
0,197530864198	15
0,275229357798	19
0,281553398058	10
0,286274509804	46
0,286458333333	10
0,681415929204	27
0,811594202899	41
0,840277777778	119
0,868421052632	23
0,888888888889	14
0,925925925926	40
0,928571428571	15
1,000000000000	23
	610

Minimum 15 cases in 1 interval	
0,126262626263	108
0,170731707317	100
0,286274509804	90
0,811594202899	78
0,840277777778	119
0,868421052632	23
0,925925925926	54
0,928571428571	15
1,000000000000	23
	610

Figuur 18 : Resultaat intervallen minimum 10 cases in 1 interval en minimum 15 cases in 1 interval

Nu we de intervallen verkregen hebben, kunnen we opnieuw de Brier score voor de verschillende methoden en voor de verschillende databestanden berekenen. Als eerste wordt de calculatie gemaakt voor de gegevens met calibratie. Om de cases in de intervallen te plaatsen, wordt er gebruikt gemaakt van de functie 'FREQUENCY'. Deze functie berekent hoe vaak de waarden binnen het vastgestelde bereik voorkomen en geeft dan een verticale array van aantallen weer, die meer dan één element hebben. Het analyseert een serie van waarden en vat deze samen in een aantal gespecificeerde ranges. Het vastgestelde bereik is hier de data die zijn weergegeven in het databestand

en in dit geval de gecalibreerde gegevens, dit zijn de variabelen met voorvoegsel 'Calibrated'.

De eerste berekening van de Brier score die gemaakt wordt, is deze voor de beslissingsboom met negen gemeenschappelijke intervallen. Daarna wordt ook voor de logistische regressie en voor de discriminant-analyse de Brier score berekend met negen gezamenlijke intervallen. De resultaten van deze calculatie zijn weergegeven in tabel 5.

Tabel 5 : Brier score met 9 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,224**	**0,227**	**0,175**
Calibration loss	**0,033**	**0,033**	**0,005**
Refinement loss	**0,191**	**0,194**	**0,170**

Uit deze tabel kunnen we afleiden dat de beslissingsboom er als beste model uitkomt. De Brier score van de decision tree is 0,175, dit is beduidend lager dan de score voor de logistische regressie en de discriminant-analyse. We kunnen vaststellen dat dit verschil zowel op te merken is in de calibration loss als in de refinement loss. Als we deze vergelijken, dan zien we dat er ongeveer een verschil van 0,025 is.

Bovendien is er een berekening gemaakt voor de gegevens waarop geen calibratie toegepast is en de resultaten van deze bewerking zijn terug te vinden in tabel 6.

Tabel 6 : Brier score met 9 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,263**	**0,331**	**0,212**
Calibration loss	**0,057**	**0,094**	**0,032**
Refinement loss	**0,206**	**0,237**	**0,180**

Wat hier onmiddellijk in het oog springt, is dat de Brier score bij alle methoden hoog ligt. De score voor de discriminant-analyse is zelfs 0,331. We zien ook dat de calibratie hier

niet goed is. De calibration loss is bij alle drie de methoden vrij hoog. Dit wil zeggen dat de voorspellingen niet zo nauwkeurig zijn. Wel kunnen we afleiden dat de beslissingsboom er als beste model uitkomt, aangezien deze methode de laagste Brier score heeft.

Zoals hierboven beschreven, zijn de calculaties ook gebeurd voor 17 gemeenschappelijke intervallen. De resultaten hiervan zijn voor databestand 1 weergegeven in de volgende twee tabellen, tabel 7 en tabel 8.

Tabel 7 : Brier score met 17 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,225**	**0,270**	**0,222**
Calibration loss	**0,027**	**0,046**	**0,050**
Refinement loss	**0,198**	**0,224**	**0,172**

Tabel 8 : Brier score met 17 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,186**	**0,203**	**0,171**
Calibration loss	**0,007**	**0,019**	**0,001**
Refinement loss	**0,179**	**0,184**	**0,170**

Het eerste dat hier opvalt, is dat de Brier score van de gegevens met calibratie veel beter is; de scores liggen hier gevoelig lager. Bovendien kunnen we opnieuw vaststellen dat de beslissingsboom er als beste methode uitkomt. Bij de data met calibratie is dit duidelijk de beste techniek, in de andere tabel leiden we af dat de decision tree maar een klein beetje beter scoort dan de logistische regressie. De calibration loss bij de gegevens waarop geen calibratie toegepast is, ligt bij de drie methoden weer vrij hoog. De classifiers voorspellen de nieuwe voorbeelden niet zo nauwkeurig. In tabel 8 is het enkel nog de discriminant-analyse die niet accuraat voorspelt. Aangezien de calibration loss voor de logistische regressie en de beslissingsboom vrij dicht bij nul liggen.

Als we naar de resultaten van de negen databestanden kijken, stellen we vast dat de Brier score over het algemeen vrij hoog ligt. Het is niet raar dat er een score van meer dan 300 te voorschijn komt, dit komt regelmatig voor en dan vooral bij de discriminant-analyse. Een verklaring voor dit fenomeen is dat er bij de discriminant-analyse vaak veel cases in eenzelfde interval zitten.

Wat er bovendien opvalt, is dat er minder intervallen zijn die geen cases bezitten. Tijdens het maken van de berekeningen viel het op, dat er vooral bij de logistische regressie en de discriminant-analyse meer intervallen waren die werkelijk observaties bevatten. Bijvoorbeeld bij databestand 7, zien we bij de resultaten van de gegevens zonder toepassing van calibratie voor de logistische regressie, dat alle elf gezamenlijke intervallen zijn opgevuld. Er is geen enkel interval dat geen observaties bevat. Dus het probleem om geen cases in een aantal intervallen te hebben, is met deze techniek gedeeltelijk opgelost. De uitkomsten van deze berekeningen zijn voor de negen databestanden terug te vinden in bijlage 2.

2. Gebruikmakend van de drie gecalibreerde probabiliteiten

Om te beginnen met deze laatste methode moeten we nogmaals weten in welke intervallen de huidige gecalibreerde probabiliteiten liggen. We maken opnieuw gebruik van de drie gecalibreerde waarschijnlijkheden per case. We controleren in hoeveel intervallen deze probabiliteiten liggen, op dezelfde manier als bij de methode met de minimum refinement loss. Dit doen we dus opnieuw door middel van X-en te plaatsen in het interval waar de waarschijnlijkheid bijhoort. Ook nu moeten we vaststellen in hoeveel verschillende intervallen de gecalibreerde waarschijnlijkheden te vinden zijn.

Een andere methode die kan toegepast worden als de drie gecalibreerde probabiliteiten in éénzelfde interval liggen, is om de case dan drie keer te gebruiken. Bijvoorbeeld, we hebben bij de eerste case als gecalibreerde waarschijnlijkheden 0; 0,126263 en 0,170732. De eerste probabiliteit valt in het interval [0; 0,076923], de tweede in interval [0,126262; 0,170731] en de laatste ligt in het volgende interval [0,170731; 0,177570]. Deze probabiliteiten liggen dus allemaal in een verschillend interval, daarom nemen we

deze drie gecalibreerde probabiliteiten mee om de gemeenschappelijke intervallen te bepalen. In het geval wanneer de waarschijnlijkheden in slechts twee intervallen zitten, maken we nogmaals gebruikt van de meerderheidsregel. Het interval waarin de meeste waarschijnlijkheden zitten, wordt mee gebruikt om de gezamenlijke intervallen te zoeken.

Nu we het aantal waarden hebben, kunnen we op zoek gaan naar de gemeenschappelijke intervallen. Het totaal aantal waarden ligt nu hoger dan de 610 die we bij de vorige methoden toegepast hebben. Het aantal getallen, dat we nu aanwenden, hangt af van de hoeveelheid cases waarvan de probabiliteiten, die gecalibreerd zijn, in twee intervallen vallen. Het maximum aantal waarden is 1830; dit betekent dat bij alle cases de huidige waarschijnlijkheden in drie verschillende intervallen zitten.

Om de overgebleven intervallen vast te stellen, wordt opnieuw de functie 'FREQUENCY' uitgevoerd. Voor databestand 1 krijgen we dan een verdeling waarin alle intervallen die er al waren, nog aanwezig zijn. Met andere woorden, er is geen enkel interval waarin er geen cases voorkomen. Aangezien we toch het aantal intervallen willen reduceren of verminderen, gaan we een norm opleggen. Deze norm bepaalt het minimum aantal cases die in het interval moeten zitten, opdat dit interval behouden wordt. Er worden twee criteria vooropgesteld. Als eerste wordt de norm op minimum 30 observaties in één interval gezet. Als we deze regel toepassen, verkrijgen we als resultaat dat er nog 20 intervallen resteren. De tweede norm zetten we op minstens 50 cases in één enkel interval en dan krijgen we als resultaat dat er voor databestand 1 nog 11 intervallen overblijven. Dit aantal wordt als voldoende beschouwd om de Brier score te berekenen. Uit figuur 19 kan men vaststellen welke intervallen er voor databestand 1 nog overblijven.

Minimum 30 cases in 1 interval
0,126262626263
0,170731707317
0,177570093458
0,197530864198
0,275229357798
0,281553398058
0,286274509804
0,286458333333
0,347826086957
0,370967741935
0,380000000000
0,500000000000
0,641975308642
0,663716814159
0,681415929204
0,744444444444
0,811594202899
0,821917808219
0,840277777778
0,925925925926

Minimum 50 cases in 1 interval
0,126262626263
0,170731707317
0,177570093458
0,281553398058
0,286274509804
0,286458333333
0,380000000000
0,663716814159
0,744444444444
0,840277777778
0,925925925926

Figuur 19 : Resultaat intervallen minimum 30 cases in 1 interval en minimum 50 cases in 1 interval

Met deze nieuwe intervallen wordt de Brier score nogmaals berekend voor de beslissingsboom, de logistische regressie en de discriminant-analyse. De eerste calculatie die er gemaakt wordt, is deze voor de drie methoden met 11 gezamenlijke intervallen en voor de gegevens met calibratie. We berekenen in het begin de calibration loss en daarna de refinement loss. Deze twee uitkomsten worden dan samengeteld tot de Brier score en geven weer hoe goed het model voorspelt. De resultaten die bij databestand 1 horen, zijn weergegeven in tabel 9.

Tabel 9 : Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,187**	**0,195**	**0,167**
Calibration loss	**0,009**	**0,010**	**0,002**
Refinement loss	**0,178**	**0,185**	**0,165**

Uit deze tabel kunnen we afleiden dat de beslissingsboom er als model bovenuit steekt. De score van de beslissingsboom ligt opvallend lager dan bij de andere twee methoden. De decision tree voorspelt de nieuwe voorbeelden nauwkeuriger dan de logistische regressie en de discriminant-analyse; dit kan vastgesteld worden door de calibration loss. Dezelfde berekeningen zijn toegepast op de gegevens waar geen calibratietechniek op uitgevoerd is. De uitkomsten van deze calculaties zijn terug te vinden in tabel 10.

Tabel 10 : Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,195**	**0,238**	**0,169**
Calibration loss	**0,009**	**0,028**	**0,002**
Refinement loss	**0,186**	**0,210**	**0,167**

Ook hier is de beslissingsboom het beste model. Bij deze gegevens is het verschil tussen de andere methoden nog groter. Dit wil zeggen dat als men moet kiezen tussen deze drie modellen, men het best kiest voor de decision tree. Wat we bovendien kunnen concluderen uit deze tabel is dat de discriminant-analyse het nog slechter doet op vlak van het nauwkeurig of accuraat voorspellen. De calibration loss bij dit model is 0,028.

De Brier scores over de negen bestanden met gegevens liggen hier lager dan als we gebruik maken van de minimum refinement loss. We zouden dus eerder opteren voor deze manier om de nieuwe, gezamenlijke intervallen te bepalen. Bovendien zijn er hier databestanden waarbij de logistische regressie er als beste methode uitkomt. De Brier score van de logistische regressie is dan kleiner dan deze voor de beslissingsboom. Dit komt onder andere voor bij databestand nummer drie als de Brier score berekend wordt

met 19 gemeenschappelijke intervallen en met de gegevens waarop calibratie toegepast is. De score voor de beslissingsboom is dan 0,184 en voor de logistische regressie is deze score 0,180; wat een lagere score is. De resultaten van deze calculaties zijn voor de negen bestanden met data te raadplegen in bijlage 3.

Hoofstuk V: Conclusies in verband met de berekeningen

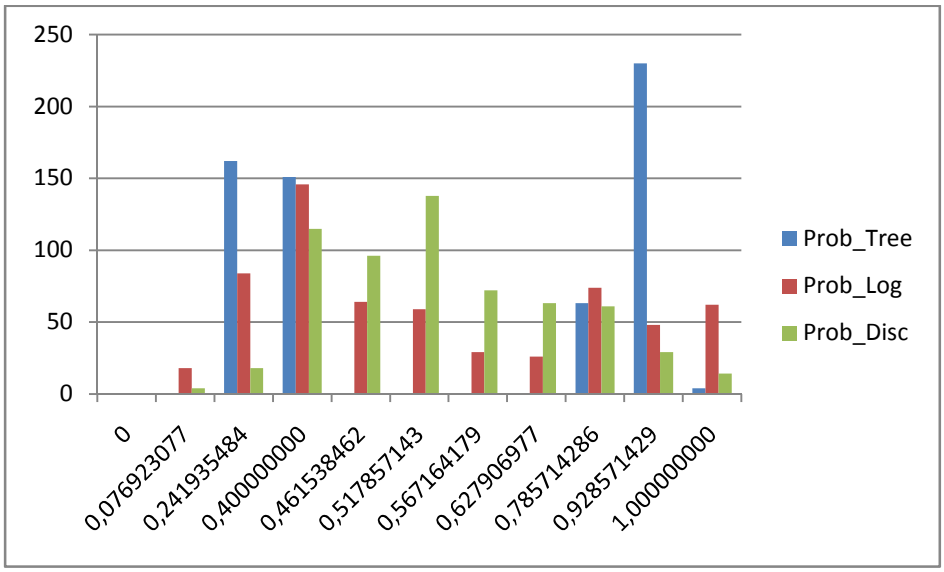
We maken gebruik van drie verschillende methoden, namelijk logistische regressie, beslissingsboom en discriminant-analyse. Ons doel is om deze drie methoden te vergelijken en eventueel zelfs te combineren. We maken eerst gebruik van de gegevens waarop geen calibratie is toegepast. We hebben al kunnen vaststellen dat de logistische regressie en de discriminant-analyse geen intervallen hebben. Ze hebben voor elke case een unieke probabiliteit. Door deze unieke waarschijnlijkheden is het onmogelijk om deze classifiers te vergelijken. Wat er ook nog opvalt, is dat de verschillende classifiers een andere verdeling hebben. Als we de verdeling van de beslissingsboom bestuderen, dan stellen we vast dat er een heleboel cases bij het begin liggen, dicht bij nul, en aan het einde, dicht bij één. De logistische regressie en de discriminant-analyse daarentegen hebben zo goed als een gewone verdeling. De cases zijn verspreid over vele verschillende intervallen, doordat elk interval een unieke waarschijnlijkheid heeft. Wel kunnen we vaststellen dat de discriminant-analyse bijna geen cases in de buurt van nul heeft; deze verdeling ligt meer naar rechts.

Als we de resultaten van de Brier score voor de gecalibreerde methoden hebben, gaan we opnieuw proberen om deze drie gecalibreerde methoden te vergelijken en te combineren. Aangezien elke methode, ook de logistische regressie en de discriminant-analyse, nu intervallen heeft, is er de mogelijkheid om deze drie modellen te combineren. De calibratietechniek zorgt ervoor dat er intervallen gegenereerd worden en geeft elk interval een unieke probabiliteit mee. Elke case wordt toegewezen aan één enkel uniek interval. Als we ze willen vergelijken naar calibratie en refinement toe, blijft dit nog steeds voor moeilijkheden zorgen. Dit is het gevolg van het verschillend aantal intervallen. Elke methode heeft een ander aantal intervallen. Zo heeft bijvoorbeeld de logistische regressie na toepassing van de calibratietechniek 17 intervallen, de beslissingsboom 6 intervallen en de discriminant-analyse heeft zelfs nog 27 intervallen. De intervallen die hier gegenereerd zijn, zijn op basis van de gegevens van databestand 1. Zo is het inderdaad moeilijk om de methoden te vergelijken. Bovendien zijn de posities van de eindpunten van deze intervallen voor de verscheidene modellen niet dezelfde. Zo kan het zijn dat ze wel hetzelfde aantal intervallen hebben, maar dat de eindpunten van deze intervallen

verschillend zijn. Bijvoorbeeld een model heeft drie intervallen: 0,1; 0,5 en 0,7. Een ander model heeft ook drie intervallen met volgende eindpunten: 0,1; 0,6 en 0,9. Ze hebben dus hetzelfde aantal intervallen, maar de posities van de eindpunten van deze intervallen liggen op een andere plaats. Dit is een extra moeilijkheid als men deze modellen wil vergelijken.

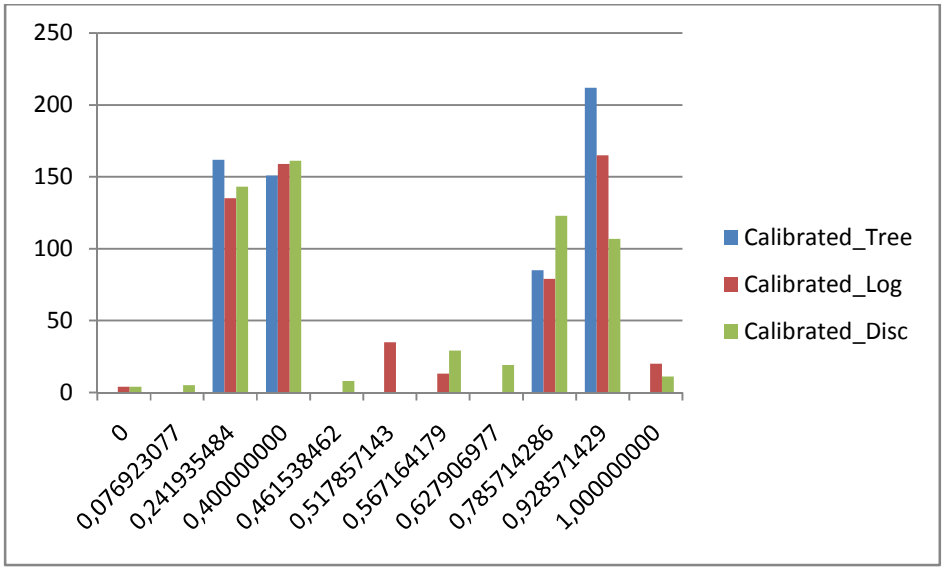
A. Methode met algoritme gemeenschappelijke intervallen

Nadat we de uitkomsten van de berekeningen met gezamenlijke intervallen voor de verschillende bestanden hebben, gaan we nogmaals proberen om de verscheidene methoden te vergelijken. Elke methode heeft nu hetzelfde aantal intervallen en bovendien zijn de posities van de eindpunten van de intervallen gelijk. Aangezien we nu gemeenschappelijke intervallen hebben, zouden we verwachten dat vergelijken van calibratie en refinement lukt. Maar doordat de verdeling van de classifiers verschilt, blijft het nog steeds moeilijk. Dit is duidelijk te zien in figuur 20 voor de gegevens waarop geen calibratie toegepast is.



Figuur 20 : Verdeling classifiers voor gegevens zonder toepassing calibratie voor databestand 1, methode 1

We kunnen uit deze grafiek afleiden dat de beslissingsboom (blauw histogram) voor een groot aantal van de intervallen geen cases heeft en dit gebeurt dan vooral bij de middelste intervallen. Bij dit model zitten de meeste cases in enkele intervallen. De logistische regressie (rood histogram) en de discriminant-analyse (groen histogram) daarentegen hebben wel in elk interval minstens één case zitten. Alleen het interval nul bezit geen enkele case. Wat ook opvalt bij de verdeling van de logistischre regressie is, dat het een links scheve verdeling is. De meeste cases bevinden zich in de lagere intervallen. Bij de verdeling van de discriminant-analyse stellen we vast dat de intervallen met de meeste cases zich in het midden bevinden. Doordat de verdeling van de verschillende methoden zo verschilt, is het moeilijk om deze classifiers te vergelijken.



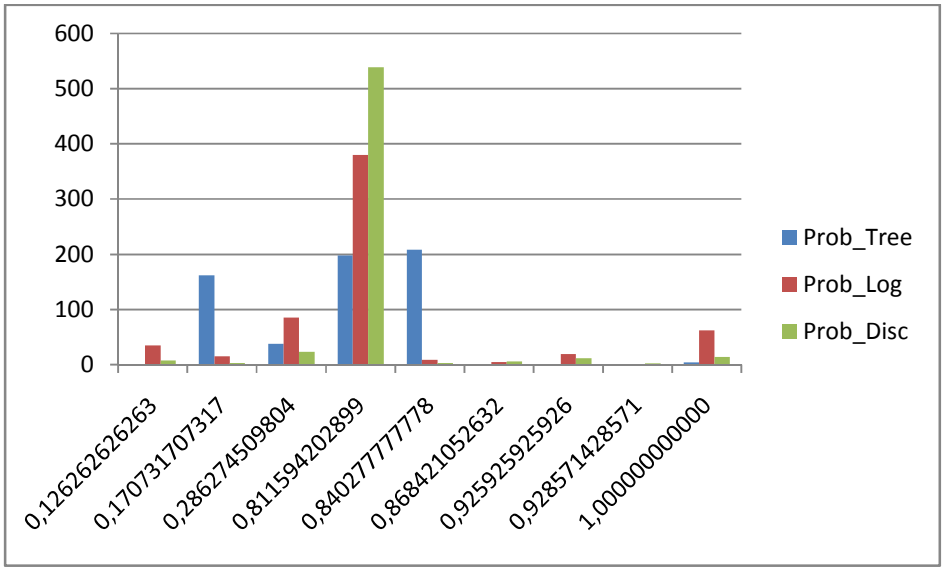
Figuur 21 : Verdeling classifiers voor gegevens met calibratie voor databestand 1, methode 1

De verdelingen van de verscheidene classifiers waarbij gebruik gemaakt wordt van de gegevens met calibratie, zijn terug te vinden in bovenstaande grafiek (zie figuur 21). We kunnen hieruit concluderen dat de verdeling voor de verschillende modellen hier beter overeenkomt dan bij de vorige grafiek. De intervallen waarin de meeste cases zich bevinden, zijn voor de drie methoden dezelfde. Wel valt het op dat de beslissingsboom nogmaals geen cases heeft in de middelste intervallen en deze keer ook niet in de

buitenste intervallen. Alle cases vallen in vier intervallen voor deze classifier. Dus het vergelijken van de verschillende classifiers gaat nog steeds moeilijk zijn.

B. Methode gebruikmakend van minimum refinement loss

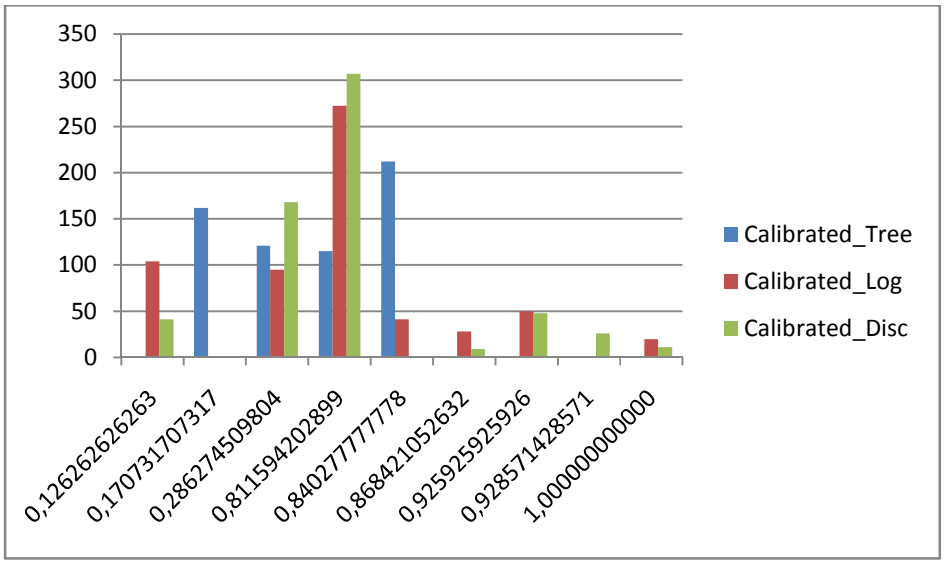
Als we de resultaten van de berekeningen hebben, willen we nogmaals proberen om de verschillende methoden te vergelijken. De modellen hebben hetzelfde aantal intervallen en ook de posities van de eindpunten van de intervallen zijn voor elke classifier dezelfde. We kunnen vaststellen dat er hier negen intervallen zijn, deze uitkomst hebben we verkregen voor databestand 1 (zie figuur 22).



Figuur 22 : Verdeling classifiers voor gegevens zonder toepassing calibratie voor databestand 1, methode 2

Opnieuw gaan we kijken naar de verdeling van de verscheidene classifiers en dit voor de gegevens waarop geen calibratietechniek uitgevoerd is. Uit figuur 22 kunnen we afleiden dat voor de logistische regressie en de discriminant-analyse het merendeel van de cases in hetzelfde interval vallen. Voor de discriminant-analyse zijn dit bijna alle cases. De beslissingsboom maakt gebruik van vier intervallen, de andere intervallen voor dit model zijn leeg. De toewijzing van de cases bij de logistische regressie is iets beter dan bij de

discriminant-analyse, maar is nog niet goed. Ook hier bevat één enkel interval de meerderheid van de observaties. Doordat deze verdelingen voor de verschillende intervallen niet gelijk zijn, is het weer moeilijk om de verscheidene modellen te vergelijken.

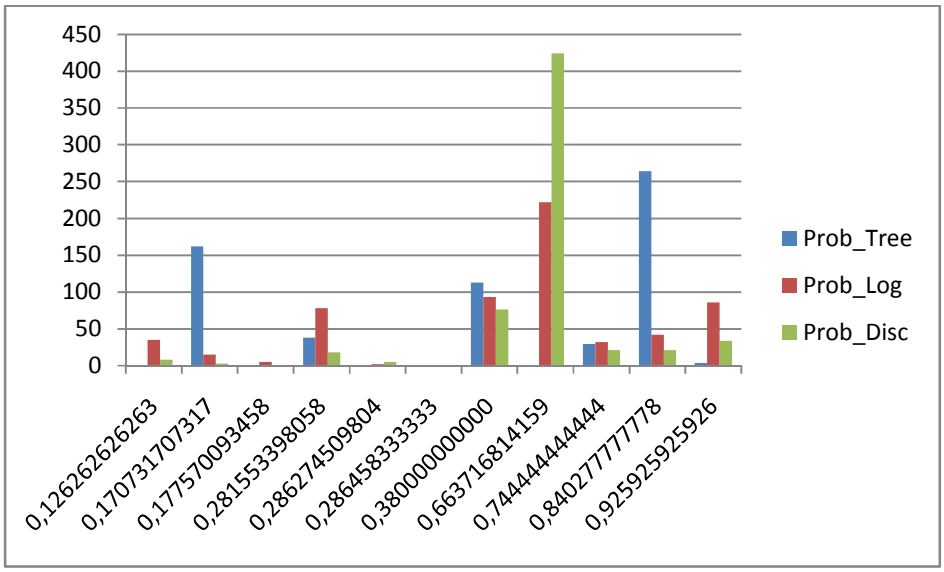


Figuur 23 : Verdeling classifiers voor gegevens met calibratie voor databestand 1, methode 2

Op de hierbovenstaande grafiek in figuur 23 vinden we de verdeling van de verschillende classifiers terug en dit voor de gegevens met calibratie. Ook hier kunnen we vaststellen dat deze verdeling voor de verscheidene modellen verschillend is. De cases van de beslissingsboom bevinden zich opnieuw in vier intervallen, de andere intervallen bevatten geen observaties. Voor de logistische regressie en de discriminant-analyse kunnen we afleiden dat de observaties beter over de verschillende intervallen verdeeld zijn. Er is geen interval meer dat de meerderheid van de cases bezit. Er zijn bovendien intervallen waarin er maar voor één classifier observaties zitten. We kunnen nogmaals concluderen dat ook via deze methode het niet makkelijk is om de verschillende classifiers te vergelijken door de verscheidene verdelingen van deze classifiers.

C. Methode gebruikmakend van drie gecalibreerde probabiliteiten

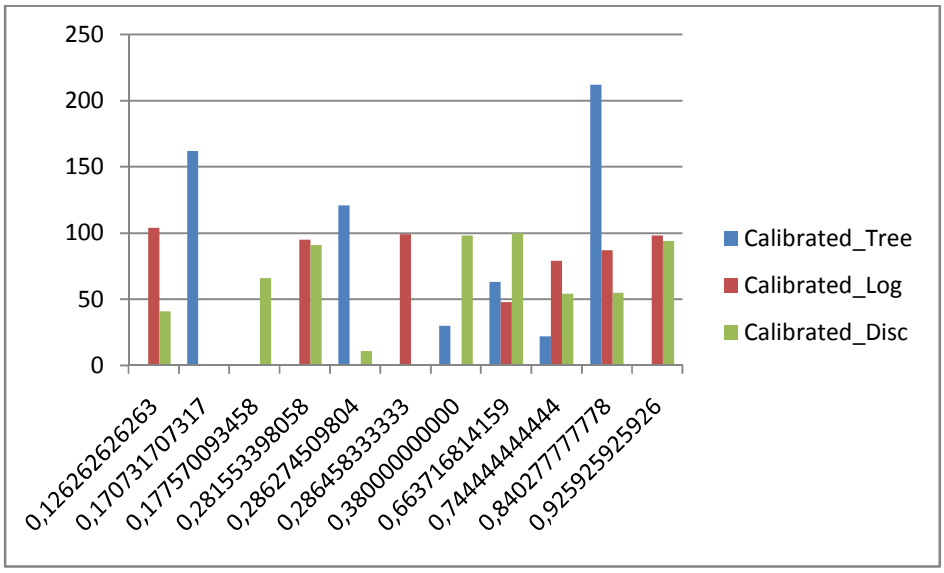
Tenslotte worden er ook conclusies opgemaakt voor de resultaten van de laatste methode, de methode met de drie gecalibreerde waarschijnlijkheden om de resterende intervallen te bepalen. Voor databestand 1 maken we hier gebruik van elf intervallen en de verdeling van de drie modellen voor de gegevens waarop calibratie niet toegepast is, vinden we terug in onderstaande figuur 24.



Figuur 24 : Verdeling classifiers voor gegevens zonder toepassing calibratie voor databestand 1, methode 3

Ten eerste is het direct duidelijk dat de meeste cases zich in de laatste intervallen bevinden en dit voor alle drie de classifiers. Bovendien is er één interval waarbij er voor elk model geen enkele observatie in zit. Voor de beslissingsboom kunnen we vaststellen dat alle cases zich in vijf intervallen bevinden, wat wil zeggen dat de rest van de intervallen leeg zijn. De observaties liggen toch nog goed verspreid over deze vijf intervallen. Voor de discriminant-analyse daarentegen bevindt de meerderheid van de cases zich in één enkel interval. In dit interval zitten meer dan 400 observaties. Voor de logistische regressie is de verdeling van de observaties iets meer verspreid, bijna elk

interval bezit toch wel een aantal cases. Er is één interval dat er een beetje bovenuitsteekt met het aantal observaties, maar over het algemeen is deze verdeling gelijkmatiger. Omdat we klaarblijkelijk uit de grafiek kunnen afleiden dat de verdelingen heel erg verschillend zijn tussen de verschillende classifiers, is het ontzettend moeilijk om deze classifiers op een goede manier te vergelijken.



Figuur 25 : Verdeling classifiers voor gegevens met calibratie voor databestand 1, methode 3

De eerste vaststelling uit deze grafiek in figuur 25 met de verdelingen van de modellen voor de gegevens waarop een calibratietechniek uitgevoerd is, is dat de cases goed verspreid liggen over de intervallen. Het interval met de meeste observaties bestaat uit iets meer dan 200 cases. Dit is toch opvallend minder dan bij de vorige grafiek. De beslissingsboom maakt hier gebruik van nog een extra interval en dit brengt het totaal aantal intervallen voor dit model op zes. De verdeling van de logistische regressie is gelijkmatig verdeeld over de resterende intervallen. Elk interval bevat ongeveer evenveel observaties. Hetzelfde patroon kunnen we min of meer ontdekken voor de discriminant-analyse. Hier is het wel zo dat er meer verschil zit tussen het aantal cases in de verschillende intervallen. Indien we deze modellen nu willen vergelijken, gaat dit moeilijk zijn. We hebben hier opnieuw dezelfde reden als bij de vorige methoden, de verdelingen

van de verschillende classifiers zijn te verschillend om een goede vergelijking mogelijk te maken.

Hoofstuk VI: Opbouw van de spreadsheets

A. Berekeningen via algoritme gemeenschappelijke intervallen

De MS Excel® bestanden waar in dit hoofdstuk naar verwezen wordt, kunnen geraadpleegd worden op de CD-rom die terug te vinden is bij deze masterproef.

Op het eerste tabblad 'Berekeningen Brier score' zijn de Brier scores van de verschillende modellen terug te vinden en dit zowel voor de gegevens die gebruik maken van de calibratietechniek als de data waarop geen calibratie toegepast is. Als eerste, zien we dat de Brier score berekend is voor de beslissingsboom met de gegevens waarop geen calibratie uitgevoerd is. Dit zien we aan het voorvoegsel 'Prob' in de cel A3 (zie figuur 29). In de cellen A4 tot en met A12 staan de probabiliteiten. Welke deze probabiliteiten zijn, hangt af van welke gegevens er als input gebruikt worden en natuurlijk is het ook afhankelijk van het model dat gebruikt wordt. Als de logistische regressie gebruikt wordt, geeft dit andere waarschijnlijkheden weer dan als de beslissingsboom zou worden toegepast. In kolom B van figuur 29 wordt het aantal waarden weergegeven dat bij die bepaalde probabiliteit hoort, die dus zoals juist beschreven in kolom A staat. De positief voorspelde waarden van dit aantal is terug te vinden in kolom C van figuur 29, die als omschrijving 'sum' heeft meegekregen. Vervolgens wordt er in cel B13 en C13 respectievelijk de som gemaakt van de waarden die zich in de cellen B4 tot B12 en C4 tot C12 bevinden. Het totaal dat in B13 staat, moet gelijk zijn aan 610, aangezien de input bestaat uit 610 cases. We kunnen dus ook vaststellen dat het aantal positief voorspelde waarden gelijk is aan de helft van het totaal aantal observaties, namelijk 305.

	A	B	C
1	Case Summaries		
2	V9		
3	Prob_Tree	N	Sum
4	0,147	162	25
5	0,242	38	9
6	0,336	83	24
7	0,350	30	13
8	0,710	29	20
9	0,750	34	22
10	0,788	22	20
11	0,818	208	168
12	1,000	4	4
13	Total	610	305

Figuur 29 : Case summaries in MS Excel®

In figuur 30 zien we de volgende kolommen voor de berekening van de calibration loss. De kolommen E tot en met G worden daarna gebruikt voor het berekenen van de calibration loss en de kolommen I tot en met K voor de refinement loss. In kolom E wordt r_j bepaald. Bijvoorbeeld cel E4 wordt bepaald door cel C4 te delen door cel B4. Dit wil zeggen dat het positief voorspelde aantal waarden gedeeld wordt door het totaal aantal waarden. Kolom F staat voor het kwadraat van het verschil tussen r_j en de bijhorende waarschijnlijkheid. De cel E4 wordt afgetrokken van de cel A4 en wordt dan vermenigvuldigd met hetzelfde verschil, (E4-A4). Daarnaast, in kolom G wordt het resultaat dat in kolom F weergegeven staat, vermenigvuldigd met het aantal dat bij die bepaalde probabiliteit hoort. Dus cel F4 wordt vermenigvuldigd met cel B4 wat het geval is voor de eerste probabiliteit. In cel F15 wordt dan de som genomen van de cellen G4 tot en met G12. Als dit allemaal berekend is, kunnen we de calibration loss bepalen. Dit gebeurt met behulp van de functie 'ROUND'. Deze functie rond een getal af naar een specifiek aantal cijfers, dat je op voorhand meegeeft. De functie wordt gebruikt in cel G15 en ziet er als volgt uit: ROUND((1/B13)*F15;3). De uitkomst van cel F15 vermenigvuldigen we met de inverse van het totaal aantal waarden dat weergegeven is in cel B13. Dit resultaat wordt dan afgerond op drie cijfers na de komma via de ROUND-functie.

	E	F	G
1			
2			
3	rj	(rj-pj)²	nj*(rj-pj)²
4	0,154320988	5,35969E-05	0,008682691
5	0,236842105	2,66039E-05	0,001010947
6	0,289156627	0,002194302	0,182127036
7	0,433333333	0,006944444	0,208333333
8	0,689655172	0,000413912	0,012003448
9	0,647058824	0,010596886	0,360294118
10	0,909090909	0,014663008	0,322586182
11	0,807692308	0,000106249	0,022099692
12	1	0	0
13			
14		Som nj*(rj-pj)²	Loss c
15		1,117137448	0,002

Figuur 30 : Berekening calibration loss in MS Excel®

Vervolgens berekenen we de refinement loss in de resterende kolommen (zie figuur 31). Het eerste wat we doen, is in kolom I het aantal waarden vermenigvuldigen met r_j, wat voor de eerste probabiliteit neerkomt op B4*E4. Hierna maken we in kolom J het verschil tussen één en de inhoud van cel E4, wat r_j is. In kolom K vermenigvuldigen we de resultaten, die we krijgen uit kolom I en J. Om de refinement loss nu te berekenen, tellen we de cellen uit kolom K met het bereik van K4 tot en met K12 op en dit gebeurt in cel J15. Daarna wordt in cel K15 opnieuw de ROUND-functie gebruikt na de vermenigvuldiging van J15 met de inverse van het totaal aantal waarden van de dataset en dit is nogmaals weergegeven in cel B13. De functie die in cel K15 wordt aangewend, is de volgende: ROUND((1/B13)*J15;3). Als laatste wordt er in cel F18 de som gemaakt van de calibration loss, G15 en de refinement loss, K15. Deze som geeft dan de uiteindelijke Brier score weer, het resultaat dat we nodig hebben. Ook hier wordt nogmaals de ROUND-functie gebruikt, ROUND(G15+K15;3).

	I	J	K
1			
2			
3	nj*rj	1-rj	(nj*rj)*(1-rj)
4	25	0,845679012	21,14197531
5	9	0,763157895	6,868421053
6	24	0,710843373	17,06024096
7	13	0,566666667	7,366666667
8	20	0,310344828	6,206896552
9	22	0,352941176	7,764705882
10	20	0,090909091	1,818181818
11	168	0,192307692	32,30769231
12	4	0	0
13			
14		Som (nj*rj)*(1-rj)	Loss r
15		100,5347806	0,165

Figuur 31 : Berekening refinement loss in MS Excel®

De uitleg die hierboven gedaan is, is enkel voor de beslissingsboom waarvan de inputgegevens geen calibratie ondergaan hebben. De berekeningen voor de gegevens met de calibratietechniek en de berekeningen voor de andere methoden gebeuren op dezelfde manier en zijn op dezelfde wijze opgebouwd.

Het tweede tabblad 'Algoritme s=0,01' zit op dezelfde wijze in mekaar als het derde en vierde tabblad. Enkel de waarde van de parameter s is verschillend. Op dit tabblad, dat gedeeltelijk in figuur 32 getoond wordt, wordt de uitvoering van het algoritme om gemeenschappelijke intervallen te krijgen, uiteengezet. In de eerste kolom, kolom A staat de input voor de lijst met split points die we nodig hebben om het algoritme toe te passen. Dit zijn de intervallen die gecreëerd worden door de calibratietechniek voor elke methode. We hebben dus drie opdelingen in intervallen, één voor elke methode. Deze drie opdelingen worden in kolom D tot één sequentie herleid. Dit gebeurt door de intervallen in een stijgende volgorde te plaatsen. Deze reeks gaan we dan gebruiken voor de uitvoering van het algoritme in de volgende kolommen, kolom G tot en met kolom N. In cel H1 staat de waarde die de merging parameter gekregen heeft. In dit geval is de waarde 0,01. Daarna gaan we in cel J3 op zoek naar het verschil van twee splitpunten,

dat kleiner is dan deze parameter s. We testen in deze cel of het resultaat dat in H3 staat, kleiner is dan de waarde die in de cel H1 staat. Als dit inderdaad kleiner is, dan komt er in deze cel 'TRUE' te staan, in het andere geval staat er 'FALSE'. Cel H3 staat voor het verschil tussen twee opeenvolgende cellen uit de reeks. Hetzelfde gebeurt er in cel L3. Alleen wordt er nu in cel N3 gecontroleerd of dit verschil groter is dan de merging parameter s. Daarna wordt de MEDIAN-functie gebruikt in cel H6. Deze functie geeft de mediaan weer ofwel het getal in het midden uit een reeks van opgegeven getallen. MEDIAN(D5:D7), hier betekent dit dat er op zoek gegaan wordt naar het middelste getal in het bereik van cel D5 tot en met D7. Als laatste wordt er gekeken of het verschil s_j en s_i groter of kleiner is dan de parameter. In cel J9 wordt er nagegaan of dit verschil kleiner is dan s en als dit zo is, wordt er de waarde 'TRUE' weergegeven. Cel J10 daarentegen staat voor de vergelijking of dit verschil groter is dan s en geeft dan ook weer de waarde 'TRUE' terug als dit inderdaad zo is. Aan de hand van deze resultaten krijgen we de uitkomst in cel H11. Hier wordt er gebruik gemaakt van de IF-functie. Deze functie controleert of een zekere voorwaarde voldaan is en geeft een bepaalde waarde terug als het waar is en een andere waarde als het niet waar is. Deze twee waarden moeten ook opgegeven worden door de gebruiker. In cel H11 staat IF(J9=TRUE; "m";"si,m,sj"), in deze cel wordt er gecontroleerd of de waarde in cel J9 gelijk is aan 'TRUE'. Als dit zo is, dan wordt de waarde m teruggegeven en in het andere geval is het de waarde s_i, m, s_j.

	G	H	I	J	K	L	M	N
1	s =	0,01						
2	s_{i+1}-s_i				s_{j+1}-s_j			
3	s_5-s_4=	0,003787879	<s	TRUE	s_7-s_6=	0,044469081	>s	TRUE
4								
5	mediaan van s_4 → s_6							
6	m =	0,125000000						
7								
8	s_j-s_i							
9	s_6-s_4=	0,005050505	<s	TRUE				
10			>s	FALSE				
11	→	m						

Figuur 32 : Uitvoering algoritme in MS Excel®

Zo gaan we door tot we aan het einde van de sequentie met split points komen en er geen verschil tussen twee opeenvolgende punten is dat kleiner is dan s. Vervolgens maken we een nieuwe lijst met punten aan de hand van de vorige berekeningen. Deze lijst is terug te vinden in kolom Q (zie figuur 33) en heeft de omschrijving 'nieuwe unieke probabiliteiten' meegekregen.

	P	Q
1		Nieuwe unieke probabiliteiten
2	1	0
3	2	0,076923077
4	3	0,100000000
5	4	0,125000000
6	5	0,174150900
7	6	0,197530864
8	7	0,227272727
9	8	0,234042553
10	9	0,241935484
11	10	0,275229358
12	11	0,285714286
13	12	0,286458333
14	13	0,300000000
15	14	0,347826087
16	15	0,375483871
17	16	0,400000000
18	17	0,461538462
19	18	0,500000000

	P	Q
1		Nieuwe unieke probabiliteiten
22	19	0,535714286
23	20	0,567164179
24	21	0,627906977
25	22	0,641975309
26	23	0,663716814
27	24	0,674418605
28	25	0,684210526
29	26	0,732142857
30	27	0,747222222
31	28	0,785714286
32	29	0,809022908
33	30	0,821917808
34	31	0,843215812
35	32	0,868421053
36	33	0,888888889
37	34	0,925925926
38	35	1

Figuur 33 : Nieuwe lijst met intervallen

De laatste twee tabbladen (Nieuwe Brier score en Nieuwe Brier score calibratie) in deze spreadsheet zijn de calculaties van de Brier score gebruikmakend van de gemeenschappelijke intervallen. Op het tabblad 'Nieuwe Brier score' wordt de Brier score voor de gegevens waarop geen calibratietechniek uitgevoerd is, berekend en dit voor de drie verschillende methoden. Per databestand zijn er twee of drie lijsten met gezamenlijke intervallen bepaald en voor elke lijst wordt de Brier score voor elke methode berekend. Deze berekening gebeurt op dezelfde wijze als op het eerste tabblad.

Op het andere tabblad worden dezelfde calculaties gemaakt, maar dan voor de gegevens met calibratie.

B. Berekeningen met minimum refinement loss

Deze spreadsheet bestaat uit zes tabbladen. Het eerste tabblad heeft als omschrijving 'Gegevens' meegekregen. De eerste drie rijen op dit tabblad zijn data die we uit de SPSS-bestanden hebben gehaald. Het zijn de gegevens waarop calibratie toegepast is, die we hier nodig hebben. Op rij 5 van dit tabblad wordt de refinement loss voor elke probabiliteit van de gecalibreerde gegevens van de discriminant-analyse berekend. De formule die hiervoor gebruikt wordt, is zoals in het vorige hoofdstuk, gelijk aan p*(1-p). Bijvoorbeeld cel B5 heeft als vergelijking dat B1 vermenigvuldigd wordt met het verschil tussen één en B1. Voor de logistische regressie en de beslissingsboom wordt op dezelfde wijze de refinement loss voor elke waarschijnlijkheid berekend. Vervolgens gebruiken we in rij 9 nogmaals een IF-functie. We gaan met deze functie testen of de drie probabiliteiten van de verschillende methoden elk in een ander interval liggen. De formule die we gebruiken, is deze IF(Intervallen!B49=3;MIN(Gegevens!B5:B7);"FOUT"). Voor de eerste drie waarschijnlijkheden wordt op het tweede tabblad 'Intervallen' gecontroleerd of cel B49 gelijk is aan drie. Als dit inderdaad het geval is, wordt de minimale waarde binnen het bereik van B5 tot en met B7 weergegeven. Met andere woorden, er wordt gezocht naar de minimum refinement loss van deze drie gecalibreerde probabiliteiten. Als deze cel niet gelijk is aan drie, wordt er 'FOUT' weergegeven in rij 9. Bovendien krijgt deze cel met 'FOUT' een rode achtergrond met de optie voorwaardelijke opmaak. Zo valt deze waarde goed op tussen alle andere waarden. We kunnen voor de eerste drie probabiliteiten vaststellen dat ze inderdaad in een ander interval zitten. Daardoor gaan we in het bereik van B5 tot en met B7 op zoek naar de minimale waarde. Dit minimum is gelijk aan de waarde die in cel B5 wordt weergegeven en is hier nul.

Daarna zijn we op zoek gegaan naar alle cellen met een rode achtergrond en de omschrijving 'FOUT'. Bij deze cellen moeten we nagaan welke twee probabiliteiten in hetzelfde interval zitten. Als we dit gevonden hebben, typen we het bijbehorende interval in de cel onder de cel waar er 'FOUT' staat. Dit wil dus zeggen dat dit interval in rij 10

wordt ingevuld. Dit hebben we nodig om de waarden te bepalen die we in de intervallen gaan invullen. Nu we de minimum refinement loss gevonden hebben, moeten we het bijbehorende interval vinden. Dit doen we in rij 11 en we maken daarbij nogmaals gebruik van een IF-functie, IF(B9=B5;B1;IF(B9=B6;B2;IF(B9="FOUT";B10;B3))). Dit is een geneste IF-functie, dit betekent dat een functie als één van de argumenten van een andere functie gebruikt wordt. In dit geval is dit opnieuw een IF-functie, maar dit kan ook een andere zijn. Als eerste gaan we testen of de waarde die in cel B9 staat dezelfde waarde is als deze die in B5 is opgegeven. Als deze twee waarden gelijk zijn, dan wordt het bijbehorende interval weergegeven en dit staat in cel B1. Indien deze twee getallen verschillend van elkaar zijn, dan wordt er een nieuwe IF-functie getest. Hier controleren we of de waarde in cel B9 gelijk is aan de waarde in cel B6 en als dit zo is, wordt de waarde uit cel B2 weergegeven. In het andere geval controleren we een laatste IF-functie. Bij deze laatste functie wordt er getest of de waarde in cel B9 gelijk is aan de tekst 'FOUT'. Als dit inderdaad het geval is, dan wordt de waarde uit cel B10 teruggegeven en anders de waarde uit cel B3. Zo zoeken we het bijhorende interval dat we nodig hebben om de overblijvende intervallen te bepalen in de volgende stap. Een voorbeeld hiervan wordt weergegeven in figuur 34.

	A	B
1	Calibrated_Disc	0
2	Calibrated_Log	0,126262626
3	Calibrated_Tree	0,170731707
4		
5	Refinement loss	0
6		0,110320375
7		0,141582391
8		
9	Minimum refinement loss	0
10		
11	**Intervallen**	0

Figuur 34 : Berekening gegevens voor reductie van intervallen in MS Excel®

De reeks van intervallen die we nu gaan gebruiken om het aantal intervallen te reduceren, is dezelfde sequentie als de lijst met split points die we nodig hadden bij het

algoritme. De waarden die we nu gevonden hebben en die beschreven staan in rij 11 worden in deze lijst ingevuld. Dit gaat door middel van de FREQUENCY-functie. Deze functie berekent hoeveel keer een waarde voorkomt in een reeks van gegevens en geeft daarna een verticale array weer. In cel B14 staat de volgende functie: {=FREQUENCY(B$11:WM$11;A14:A61)}. Hier worden dus de gegevens uit B11 tot WM11 in de intervallen geplaatst die zich bevinden in de cellen A14 tot en met A61. Deze output wordt verder gebruikt in een ander tabblad.

Op het tweede tabblad van deze spreadsheet wordt er nagegaan in welke intervallen de drie gecalibreerde probabiliteiten zich bevinden. Dit gebeurt door gebruik te maken van een functie die iets moeilijker is. De volgende functie hoort bij cel B1 op het tabblad 'Intervallen':
IF(OR(AND(Gegevens!B$1>=Intervallen!$A1;Gegevens!B$1<Intervallen!$A2);AND(Gegevens!B$2>=Intervallen!$A1;Gegevens!B$2<Intervallen!$A2);AND(Gegevens!B$3>=Intervallen!$A1;Gegevens!B$3<Intervallen!$A2));"X";" "). Het doel van deze functie is dat er een X geplaatst wordt in het interval waar de bepaalde gecalibreerde waarschijnlijkheid toe behoort. Als het niet bij dit bepaalde interval hoort, dan komt er niets te staan in de betreffende cel. Wat er nu nog rest, is de conditie waaraan men moet voldoen. We kunnen zien in de formule dat er een OR-functie in zit, dit betekent dat er aan één van de condities voldaan moet zijn. Eerst wordt er nagegaan of de gecalibreerde probabiliteit van de discriminant-analyse (cel B1 op tabblad Gegevens) groter of gelijk is aan de waarde in cel A1 en of de waarschijnlijkheid bovendien kleiner is dan de waarde in cel A2. Daarna gebeurt hetzelfde voor de gecalibreerde waarschijnlijkheden van de logistische regressie en de beslissingsboom. Dus als één van deze condities waar is, dan wordt er een X in de betreffende cel geplaatst, anders wordt er niets geplaatst.

De formule die we terugvinden in cel B48 is iets anders dan de vorige en is gelijk aan IF(OR(Gegevens!B$1=Intervallen!$A48;Gegevens!B$2=Intervallen!$A48;Gegevens!B$3=Intervallen!$A48);"X";" "). Hier wordt er nagegaan of de gecalibreerde probabiliteit gelijk is aan de waarde in cel A48. Dit gebeurt op deze manier aangezien de waarschijnlijkheid niet kleiner kan zijn dan de volgende waarde van het interval, omdat dit het laatste interval is.

COUNTIF(B1:B48;"X") wordt gebruikt in cel B49. Deze functie sommeert het aantal cellen binnen een bepaald bereik dat aan de opgegeven conditie voldoet. Hier worden de cellen waar er een X staat, opgeteld. Zo kunnen we nagaan of de drie gecalibreerde waarschijnlijkheden elk in een ander interval liggen of niet. Deze output wordt, zoals hierboven beschreven, gebruikt voor een vergelijking op het tabblad 'Gegevens'.

Het tabblad 'Overgebleven intervallen' wordt aangewend om weer te geven welke intervallen er nog resteren na de reductie. In kolom A staan alleen de intervallen die minimaal één case bevatten, de intervallen met nul cases zijn hier al weggelaten. We hebben de getallen van het tabblad 'Gegevens' gekopieerd en de intervallen zonder cases hebben we uit deze lijst verwijderd. Onder deze reeks in cel B38 maken we de som van al deze cases om na te gaan of er nog wel degelijk 610 cases aanwezig zijn. Hiervoor gebruiken we de SUM-functie. Op dit tabblad vinden we bovendien de intervallen die nog resteren , nadat we een norm hebben opgelegd, terug. Kolom D bestaat uit deze intervallen waar er minimaal tien cases in één interval zitten. Deze getallen zijn gewoon gekopieerd uit de lijst in kolom A. In kolom E zijn de waarden die bij elk interval horen, geplaatst. Ofwel is dit dezelfde waarde als die we in kolom B terugvinden, ofwel is dit de som van meerdere waarden uit kolom B. We doen hetzelfde voor kolom G en H, maar hier is de standaard dat er minstens vijftien cases zich in één enkel interval moeten bevinden. Dit als gevolg van het te grote aantal intervallen bij de vorige norm.

Voor we verder kunnen met de berekeningen van de Brier score, hebben we nog input nodig. Deze input staat op tabblad 'Gegevens Brier score'. Hier staan data op, die we weer uit de SPSS-bestanden hebben gehaald. Dit zijn de oorspronkelijke probabiliteiten van elke methode, dus zowel van de logistische regressie, de discriminant-analyse als van de decision tree. Voor elk model hebben we de gecalibreerde waarschijnlijkheden en de probabiliteiten zonder toepassing van de calibratietechniek genomen. Elke kolom heeft ook zijn bijpassend label gekregen, zodat het duidelijk is welke gegevens in welke kolom staan. Zoals in een vorig hoofdstuk besproken is, zijn de gegevens met label 'Calibrated' de gecalibreerde probabiliteiten en de data met 'Prob' de probabiliteiten waarop geen calibratie toegepast is.

Op de twee laatste tabbladen 'Berekeningen Brier score cal' en 'Berekeningen Brier score prob' worden de eigenlijke calculaties gemaakt. Van deze calculaties hebben we de resultaten nodig om tot conclusies te komen. De berekeningen op deze tabbladen gebeuren op dezelfde wijze als hierboven beschreven werd voor de berekeningen via het algoritme. Eerst berekenen we de calibration loss en de refinement loss en daaruit maken we de som voor de Brier score. Het enige dat hier anders is, is dat we de data in de intervallen met behulp van de FREQUENCY-functie indelen. Dit gebeurde niet bij de eerste methode, de methode waar er gewerkt wordt met het algoritme.

C. Berekeningen gebruikmakend van drie gecalibreerde probabiliteiten

De spreadsheet die voor deze methode gebruikt is, bestaat uit evenveel tabbladen als bij de vorige methode, namelijk zes. De tabbladen hebben zelfs dezelfde labels meegekregen. Op het eerste tabblad 'Gegevens' staan opnieuw de data die we nodig hebben om de intervallen te gaan verminderen. Deze keer gebeurt dit wel op een andere manier.

Op rij 1 tot en met rij 3 staan de gecalibreerde waarschijnlijkheden van de drie verschillende methoden. Dit zijn dezelfde gegevens als bij de vorige methode om de intervallen te reduceren. Op het tweede tabblad 'Intervallen' gaan we verifiëren of de drie gecalibreerde probabiliteiten elk in een ander interval zitten. Dit gebeurt door middel van dezelfde formule als bij de vorige methode. Er worden opnieuw X-en geplaatst in het interval waartoe de probabiliteit behoort. In rij 49 wordt het aantal intervallen berekend waar een X in staat, deze output wordt dadelijk gebruikt voor een andere functie op het tabblad 'Gegevens'.

In de rijen 5, 6 en 7 van het tabblad 'Gegevens' gaan we de hierboven omschreven output aanwenden. Hiervoor gebruiken we nogmaals een IF-functie, in rij 5 is dit voor kolom B de volgende: (Intervallen!B$49=3;Gegevens!B1;"FOUT"). We controleren met deze functie of de waarde in cel B49 op het tabblad 'Intervallen' gelijk is aan drie, dus dat

alle gecalibreerde waarschijnlijkheden zich in een ander interval bevinden. Indien deze waarde inderdaad overeenkomt met drie, geeft deze functie de waarde weer die zich in cel B1 van dit tabblad bevindt. In het andere geval wordt er de omschrijving 'FOUT' in de cel ingevuld. Hetzelfde gebeurt voor rij 6 en rij 7, maar als de waarde nu gelijk is aan drie, wordt er respectievelijk de waarde van cel B2 en de waarde van cel B3 weergegeven.

Om de resterende intervallen te bepalen, hebben we een reeks met waarden nodig. Op dit moment hebben we vanaf kolom B, in elke kolom de drie gecalibreerde probabiliteiten staan ofwel de omschrijving 'FOUT'. Het is nu de bedoeling dat we een lijst gaan creëren met alle waarden onder elkaar. Dit doen we door middel van de volgende formule: INDEX($B$5:$WM$7;IF(MOD($A11;3)=0;3;IF(MOD($A11;3)=1;1;2));$C11) en deze exacte formule vinden we terug in cel B11. Deze formule zal verduidelijkt worden in verschillende stappen.

We maken hier gebruik van een INDEX-functie, deze functie resulteert in een waarde of de verwijzing naar een waarde vanuit een tabel of reeks. Het geeft als resultaat de verwijzing naar de cel op het snijpunt van een bepaalde rij en kolom. Als eerste zoeken we een getal op in het bereik van B5 tot en met WM7, dit is de tabel met de drie gecalibreerde waarschijnlijkheden. Om dit voor mekaar te krijgen, hebben we een rijnummer en een kolomnummer nodig. Om het rijnummer te bepalen, wordt er gebruik gemaakt van een geneste IF-functie met een MOD-functie, IF(MOD($A11;3)=0;3;IF(MOD($A11;3)=1;1;2). De MOD-functie geeft de restwaarde terug nadat een getal gedeeld wordt door de opgegeven deler. Hier gebruiken we de waarden die in kolom A staan en deze waarden bestaan uit een reeks van opeenvolgende getallen, te beginnen bij één. Het eerste getal moet uit rij 5 komen, het tweede uit rij 6, het derde uit rij 7, het vierde opnieuw uit rij 5, enz. We gaan dus met behulp van de MOD-functie het getal dat in kolom A staat, delen door drie. De restwaarde die deze functie teruggeeft, ligt altijd tussen één en drie. Daarna benutten we de IF-functie. Als de rest gelijk is aan nul, dan moet het rijnummer gelijk zijn aan drie. Indien de restwaarde overeenstemt met één, dan moet het rijnummer gelijk zijn aan één. In elk ander geval is

het rijnummer gelijk aan twee. Op deze manier bekomen we het rijnummer dat we nodig hebben.

Om het kolomnummer te bepalen, gebruiken we twee extra hulpkolommen, kolom C en D. In kolom D maken we een reeks die start bij 1/3 en n = (n-1) + 1/3. Dit gebeurt op deze manier omdat er per drie getallen het kolomnummer met één moet opschuiven. Vervolgens wordt het getal dat in kolom D staat naar boven afgerond in kolom C en wel door volgende formule: ROUNDUP(D11;0). Zo wordt het kolomnummer altijd +1 wanneer we een getal hebben dat vlak na een veelvoud van drie komt. De waarden komen nu in een verticale lijst terecht en kunnen bijna gebruikt worden om de resterende intervallen te bepalen. De bijbehorende cellen zijn terug te vinden in figuur 35.

	A	B	C	D
10				0
11	1	0	1	0,333333
12	2	0,126263	1	0,666667
13	3	0,170732	1	1
14	4	0,076923	2	1,333333
15	5	0,286458	2	1,666667
16	6	0,170732	2	2

Figuur 35 : Berekening verticale lijst met waarden in MS Excel®

De waarden uit de verticale lijst worden gekopieerd naar kolom F, vanaf cel F11. Het enige wat we nu nog moeten doen, is de cellen waar 'FOUT' in staat, vervangen door het juiste, bijbehorende interval. Als er 'FOUT' in de cel staat, moeten we op zoek gaan naar het interval waarin er twee cases zitten. Dit wordt dan ingevuld in de reeks in plaats van de omschrijving 'FOUT'. Op dit moment kunnen we de overgebleven intervallen gaan vaststellen.

De resterende intervallen worden berekend op tabblad 'Overgebleven intervallen' en dit gebeurt door middel van de FREQUENCY-functie. Dit gaat op dezelfde wijze als bij de vorige berekeningsmethode. Om de Brier score te bepalen, hebben we ook dezelfde input nodig en die wordt weergegeven op tabblad 'Gegevens Brier score'. Daarna berekenen we

deze score voor de gegevens met calibratie en voor de gegevens waarop geen calibratie toegepast is. Deze calculaties doen we nogmaals op dezelfde manier als bij de vorige methode.

Hoofdstuk VII: Besluit

Om deze masterproef af te sluiten, som ik de belangrijkste vaststellingen op uit het onderzoek en de uitgewerkte berekeningen.

In dit onderzoek wordt er nagegaan of we de verschillende classifiers met elkaar kunnen vergelijken. Het probleem met het vergelijken van deze verschillende modellen was dat het aantal intervallen niet overeenkwam en ook de posities van de eindpunten van deze intervallen niet dezelfde waren. Deze moeilijkheid hebben we proberen op te lossen met behulp van een algoritme om het aantal intervallen te reduceren. Elke classifier heeft daarna hetzelfde aantal intervallen en bovendien komen de posities van de eindpunten overeen. Bij deze uitwerking was er een nieuwe moeilijkheid die tevoorschijn kwam. We hadden nu een paar intervallen, die geen cases bevatten. Dit bemoeilijkte de berekening van de Brier score. Er werd geprobeerd om dit probleem weg te werken door middel van twee andere methoden om het totaal aantal intervallen te verminderen. Dit is gedeeltelijk gelukt, aangezien er meer intervallen die wel degelijk observaties bevatten, aanwezig zijn. Toch blijven er nog steeds intervallen waarin geen cases terug te vinden zijn.

Doordat we hetzelfde aantal intervallen en dezelfde posities van eindpunten verkregen, zouden we vermoeden dat we nu de verschillende classifiers wel zouden kunnen vergelijken. Dit is echter niet het geval en dit komt vooral doordat de verdeling van de verscheidene modellen niet dezelfde zijn. Zo zijn er sommige classifiers met een paar intervallen, die geen cases bevatten waar een andere classifier er wel heeft. Bovendien zijn er modellen waar één enkel interval het merendeel van de observaties bezit. Dit bemoeilijkt uiteraard de vergelijking tussen de modellen.

We kunnen dus vaststellen dat niet enkel de posities van de eindpunten van de intervallen belangrijk zijn, maar ook de toewijzing van de cases aan de intervallen. Indien elk interval voor elke classifier observaties bevat, kunnen we wel de verschillende modellen vergelijken en de beste classifier eruit kiezen. Er moet dus een manier gevonden worden, zodat de observaties gelijkmatig verdeeld worden over de

verschillende intervallen. Dit moet op zo'n wijze gebeuren, dat er geen enkel interval is, dat een meerderheid van de cases bevat.

Een suggestie voor verder onderzoek heeft te maken met de calibratietechniek die er gebruikt wordt. We hebben hier gebruik gemaakt van het PAV algoritme. Om de gecalibreerde gegevens te verkrijgen, wordt het algoritme uitgevoerd met als input de data die we uit de verschillende bestanden halen. Nu kunnen we proberen om het PAV algoritme aan te passen. Een aanbeveling die we hierbij kunnen doen, is om het PAV algoritme zo toe te passen dat we ook gebruik maken van de lijst van intervallen, die we verkregen hebben. Deze reeks verkrijgen we door de opdeling in intervallen van de verschillende gecalibreerde classifiers samen te nemen tot één reeks. Dus als input hebben we niet enkel de data uit de SPSS-bestanden, maar ook de sequentie van intervallen. Door dit toe te passen, kunnen we misschien wel de verscheidene modellen vergelijken en zo de beste methode eruit halen.

Tenslotte komen we nog even terug op de centrale onderzoeksvraag en de bijhorende deelvragen die in het begin van deze masterproef geformuleerd zijn. De onderzoeksvraag is: 'Wat is het effect en nut van een nieuwe calibratiemethode gebaseerd op meerdere modellen?' Om hier een antwoord op te vinden, is deze vraag opgesplitst in twee deelvragen.

-Wat zijn de voordelen van deze nieuwe calibratiemethode?

Een voordeel van deze nieuwe calibratiemethode is dat de verschillende modellen hetzelfde aantal intervallen hebben en dat de posities van deze intervallen overeenkomen.

-Wat zijn de nadelen/beperkingen van deze nieuwe calibratiemethode?

Een eerste beperking die we gevonden hebben, is dat er verscheidene intervallen zijn, die geen cases bevatten. Daarnaast is het ook van belang dat de toewijzing van de cases op een gelijkmatige manier over de verschillende intervallen gebeurt.

Lijst geraadpleegde werken

Baesens, B., Mues, C., & Vanthienen, J. (2003). Knowledge discovery in data: naar perfomante en begrijpelijke modellen van bedrijfsintelligentie [Elektronische versie]. *Business inzicht*, 12, 1,4

Batra, R., Lenk, P., & Wedel, M. (2006). *Separating brand from category personality.* Opgevraagd op 11 maart 2009, van de volgende website: http://www.scribd.com/doc/3036463/Separating-Brand-from-Category-Personality

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability [Elektronische versie]. *Monthly weather review,* Vol 78, No. 1, 1-3

Caruana, R., & Niculescu-Mizil, A. (2004). Data mining in metric space : an empirical analysis of supervised learning performance criteria [Elektronische versie]. *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining,* 69-78.

Cieslak, D. A., & Chawla, N. V. (2006). *Calibration and power of probability estimation trees on unbalanced data.* Opgevraagd op 16 maart 2009, van de volgende website: http://www.cse.nd.edu/Reports/2006/TR-2006-12.pdf

Fawcett, T. & Niculescu-Mizil, A. (2007). PAV and the ROC convex hull [Elektronische versie]. *Machine learning,* Vol 68, 97-106

Fernandez, G. C. J. (2002). *Discriminant analysis, a powerful classification technique in data mining.* Opgevraagd op 20 oktober 2008, van de volgende website: http://www2.sas.com/proceedings/sugi27/p247-27.pdf

Ferri, C., Hernández-Orallo, J., & Modroiu, R. (2009). An experimental comparison of performance measures for classification [Elektronische versie]. *Pattern recognition letters, 30,* 27-38.

Flach, P. & Matsubara, E. T. (2007). A simple lexicographic ranker and probability estimator. In *Machine Learning: ECML 2007* (pp. 575-582). Berlin / Heidelberg: Springer

Huang, J., & Ling, C. X. (2007). Constructing new and better evaluation measures for machine learning [Elektronische versie]. *Proceedings of the twentieth international joint conference on artificial intelligence,* 859-864

Jakulin, A. & Bratko, I. (2003). Analyzing attribute dependencies. In *Knowledge discovery in databases: PKDD 2003* (pp. 229-240). Berlin / Heidelberg: Springer

Ludl, M. C. & Widmer, G. (2000). Relative unsupervised discretization for association rule mining. In *Principles of data mining and knowledge discovery* (pp. 148-158). Berlin / Heidelberg: Springer.

Niculescu-Mizil, A., & Caruana, R. (2005). Predicting good probabilities with supervised leraning [Elektronische versie]. *Proceedings of the 22nd international conference on machine learning,* 625-632

Vuk, M. & Curk, T. (2006). ROC curve, lift chart and calibration plot [Elektronische versie]. *Metodoloski zvezki*, Vol 3, No. 1, 89-108

Witten, I. H., & Frank, E. (2000). *Data mining: practical machine learning tools and techniques with java implementations.* San Francisco, Californië: Morgan Kaufman

Zadrozny, B. & Elkan, C. (2001). Obtaining calibrated probability estimates from decision trees and naïve Bayesian classifiers [Elektronische versie]. *Proceedings of the eighteenth international conference on machine learning,* 609-616

Bijlagen

1. Resultaten van de 9 bestanden gebruikmakend van het algoritme
2. Resultaten van de 9 bestanden gebruikmakend van de minimum refinement loss
3. Resultaten van de 9 bestanden gebruikmakend van de 3 gecalibreerde probabiliteiten

Bijlage 1: Resultaten van de 9 bestanden gebruikmakend van het algoritme

Databestand 1

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,185**	**0,207**	**0,167**
Calibration loss	/	/	**0,002**
Refinement loss	/	/	**0,165**

Brier score met 11 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,190**	**0,210**	**0,176**
Calibration loss	**0,011**	**0,018**	**0,010**
Refinement loss	**0,179**	**0,192**	**0,166**

Brier score met 35 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,187**	**0,208**	**0,167**
Calibration loss	**0,016**	**0,025**	**0,002**
Refinement loss	**0,171**	**0,183**	**0,165**

Brier score met calibratie

	Log	Disc	Tree
Brier score	**0,182**	**0,193**	**0,168**
Calibration loss	**0,005**	**0,013**	**0,003**
Refinement loss	**0,177**	**0,180**	**0,165**

Brier score met 11 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,190**	**0,200**	**0,178**
Calibration loss	**0,009**	**0,007**	**0,010**
Refinement loss	**0,181**	**0,193**	**0,168**

Brier score met 35 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,182**	**0,194**	**0,168**
Calibration loss	**0,005**	**0,013**	**0,003**
Refinement loss	**0,177**	**0,181**	**0,165**

Databestand 2

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,162**	**0,190**	**0,152**
Calibration loss	/	/	**0,005**
Refinement loss	/	/	**0,147**

Brier score met 17 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,162**	**0,191**	**0,153**
Calibration loss	**0,011**	**0,030**	**0,005**
Refinement loss	**0,151**	**0,161**	**0,148**

Brier score met 24 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,163**	**0,190**	**0,152**
Calibration loss	**0,012**	**0,030**	**0,005**
Refinement loss	**0,151**	**0,160**	**0,147**

Brier score met 34 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,162**	**0,189**	**0,152**
Calibration loss	**0,014**	**0,032**	**0,005**
Refinement loss	**0,148**	**0,157**	**0,147**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,156**	**0,165**	**0,152**
Calibration loss	**0,006**	**0,006**	**0,004**
Refinement loss	**0,150**	**0,159**	**0,148**

Brier score met 17 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,155**	**0,167**	**0,156**
Calibration loss	**0,005**	**0,007**	**0,008**
Refinement loss	**0,150**	**0,160**	**0,148**

Brier score met 24 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,156**	**0,165**	**0,157**
Calibration loss	**0,006**	**0,004**	**0,005**
Refinement loss	**0,15**	**0,161**	**0,152**

Brier score met 34 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,156**	**0,165**	**0,153**
Calibration loss	**0,006**	**0,005**	**0,005**
Refinement loss	**0,150**	**0,160**	**0,148**

Databestand 3

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,179**	**0,202**	**0,186**
Calibration loss	/	/	**0,008**
Refinement loss	/	/	**0,178**

Brier score met 7 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,183**	**0,215**	**0,196**
Calibration loss	**0,011**	**0,017**	**0,012**
Refinement loss	**0,172**	**0,198**	**0,184**

Brier score met 10 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,183**	**0,204**	**0,192**
Calibration loss	**0,011**	**0,018**	**0,012**
Refinement loss	**0,172**	**0,186**	**0,180**

Brier score met 32 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,179**	**0,203**	**0,186**
Calibration loss	**0,013**	**0,022**	**0,008**
Refinement loss	**0,166**	**0,181**	**0,178**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,179**	**0,189**	**0,184**
Calibration loss	**0,006**	**0,008**	**0,002**
Refinement loss	**0,173**	**0,181**	**0,182**

Brier score met 7 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,186**	**0,203**	**0,191**
Calibration loss	**0,009**	**0,011**	**0,008**
Refinement loss	**0,177**	**0,192**	**0,183**

Brier score met 10 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,185**	**0,191**	**0,192**
Calibration loss	**0,008**	**0,003**	**0,009**
Refinement loss	**0,177**	**0,188**	**0,183**

Brier score met 32 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,179**	**0,189**	**0,184**
Calibration loss	**0,006**	**0,007**	**0,002**
Refinement loss	**0,173**	**0,182**	**0,182**

Databestand 4

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,172**	**0,195**	**0,167**
Calibration loss	/	/	**0,002**
Refinement loss	/	/	**0,165**

Brier score met 11 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,180	0,210	0,177
Calibration loss	0,017	0,028	0,009
Refinement loss	0,163	0,182	0,168

Brier score met 15 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,171	0,197	0,170
Calibration loss	0,010	0,022	0,003
Refinement loss	0,161	0,175	0,167

Brier score met 35 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,170	0,195	0,168
Calibration loss	0,011	0,024	0,002
Refinement loss	0,159	0,171	0,166

Brier score met calibratie

	Log	Disc	Tree
Brier score	0,168	0,175	0,168
Calibration loss	0,007	0,006	0,002
Refinement loss	0,161	0,169	0,166

Brier score met 11 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,175**	**0,178**	**0,173**
Calibration loss	**0,009**	**0,006**	**0,003**
Refinement loss	**0,166**	**0,172**	**0,170**

Brier score met 25 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,169**	**0,178**	**0,168**
Calibration loss	**0,004**	**0,006**	**0,002**
Refinement loss	**0,165**	**0,172**	**0,166**

Brier score met 35 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,168**	**0,175**	**0,168**
Calibration loss	**0,007**	**0,006**	**0,002**
Refinement loss	**0,161**	**0,169**	**0,166**

Databestand 5

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,171**	**0,200**	**0,167**
Calibration loss	/	/	**0,003**
Refinement loss	/	/	**0,164**

Brier score met 21 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,181**	**0,217**	**0,171**
Calibration loss	**0,015**	**0,035**	**0,007**
Refinement loss	**0,166**	**0,182**	**0,164**

Brier score met 26 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,190**	**0,224**	**0,187**
Calibration loss	**0,023**	**0,041**	**0,021**
Refinement loss	**0,167**	**0,183**	**0,166**

Brier score met 37 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,171**	**0,198**	**0,169**
Calibration loss	**0,017**	**0,029**	**0,005**
Refinement loss	**0,154**	**0,169**	**0,164**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,165**	**0,178**	**0,167**
Calibration loss	**0,009**	**0,008**	**0,002**
Refinement loss	**0,156**	**0,170**	**0,165**

Brier score met 21 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,170**	**0,186**	**0,171**
Calibration loss	**0,011**	**0,013**	**0,006**
Refinement loss	**0,159**	**0,173**	**0,165**

Brier score met 26 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,176**	**0,185**	**0,175**
Calibration loss	**0,017**	**0,013**	**0,008**
Refinement loss	**0,159**	**0,172**	**0,167**

Brier score met 37 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,165**	**0,179**	**0,167**
Calibration loss	**0,009**	**0,008**	**0,002**
Refinement loss	**0,156**	**0,171**	**0,165**

Databestand 6

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,178**	**0,197**	**0,165**
Calibration loss	/	/	**0,001**
Refinement loss	/	/	**0,164**

Brier score met 9 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,193**	**0,225**	**0,178**
Calibration loss	**0,010**	**0,020**	**0,012**
Refinement loss	**0,183**	**0,205**	**0,166**

Brier score met 17 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,180**	**0,199**	**0,171**
Calibration loss	**0,009**	**0,020**	**0,005**
Refinement loss	**0,171**	**0,179**	**0,166**

Brier score met 39 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,180**	**0,198**	**0,167**
Calibration loss	**0,016**	**0,027**	**0,003**
Refinement loss	**0,164**	**0,171**	**0,164**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,174**	**0,178**	**0,165**
Calibration loss	**0,007**	**0,009**	**0,001**
Refinement loss	**0,167**	**0,169**	**0,164**

Brier score met 9 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,182**	**0,197**	**0,178**
Calibration loss	**0,007**	**0,013**	**0,008**
Refinement loss	**0,175**	**0,184**	**0,170**

Brier score met 17 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,176**	**0,182**	**0,168**
Calibration loss	**0,005**	**0,006**	**0,004**
Refinement loss	**0,171**	**0,176**	**0,164**

Brier score met 39 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,174**	**0,179**	**0,166**
Calibration loss	**0,007**	**0,008**	**0,002**
Refinement loss	**0,167**	**0,171**	**0,164**

Databestand 7

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,174**	**0,200**	**0,168**
Calibration loss	/	/	**0,004**
Refinement loss	/	/	**0,164**

Brier score met 15 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,174**	**0,201**	**0,168**
Calibration loss	**0,011**	**0,021**	**0,003**
Refinement loss	**0,163**	**0,180**	**0,165**

Brier score met 29 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,175**	**0,202**	**0,168**
Calibration loss	**0,013**	**0,024**	**0,003**
Refinement loss	**0,162**	**0,178**	**0,165**

Brier score met 35 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,174**	**0,202**	**0,168**
Calibration loss	**0,013**	**0,026**	**0,003**
Refinement loss	**0,161**	**0,176**	**0,165**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,173**	**0,183**	**0,167**
Calibration loss	**0,006**	**0,008**	**0,002**
Refinement loss	**0,167**	**0,175**	**0,165**

Brier score met 15 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,173**	**0,183**	**0,167**
Calibration loss	**0,005**	**0,005**	**0,002**
Refinement loss	**0,168**	**0,178**	**0,165**

Brier score met 29 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,174**	**0,184**	**0,167**
Calibration loss	**0,004**	**0,006**	**0,002**
Refinement loss	**0,170**	**0,178**	**0,165**

Brier score met 35 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,173**	**0,183**	**0,169**
Calibration loss	**0,006**	**0,007**	**0,004**
Refinement loss	**0,167**	**0,176**	**0,165**

Databestand 9

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,189**	**0,204**	**0,181**
Calibration loss	/	/	**0,005**
Refinement loss	/	/	**0,176**

Brier score met 13 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,193**	**0,205**	**0,187**
Calibration loss	**0,012**	**0,018**	**0,007**
Refinement loss	**0,181**	**0,187**	**0,180**

Brier score met 21 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,189**	**0,205**	**0,186**
Calibration loss	**0,010**	**0,019**	**0,007**
Refinement loss	**0,179**	**0,186**	**0,179**

Brier score met 35 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,188**	**0,205**	**0,183**
Calibration loss	**0,011**	**0,020**	**0,004**
Refinement loss	**0,177**	**0,185**	**0,179**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,183**	**0,194**	**0,182**
Calibration loss	**0,006**	**0,009**	**0,004**
Refinement loss	**0,177**	**0,185**	**0,178**

Brier score met 13 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,190**	**0,195**	**0,192**
Calibration loss	**0,009**	**0,008**	**0,013**
Refinement loss	**0,181**	**0,187**	**0,179**

Brier score met 21 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,186**	**0,198**	**0,184**
Calibration loss	**0,006**	**0,010**	**0,006**
Refinement loss	**0,180**	**0,188**	**0,178**

Brier score met 35 vaste intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,183**	**0,195**	**0,183**
Calibration loss	**0,006**	**0,009**	**0,005**
Refinement loss	**0,177**	**0,186**	**0,178**

Databestand 10

Brier score van gegevens waarop geen calibratietechniek is toegepast

	Log	Disc	Tree
Brier score	**0,174**	**0,196**	**0,166**
Calibration loss	/	/	**0,002**
Refinement loss	/	/	**0,164**

Brier score met 8 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,185**	**0,204**	**0,181**
Calibration loss	**0,018**	**0,022**	**0,015**
Refinement loss	**0,167**	**0,182**	**0,166**

Brier score met 11 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,183**	**0,201**	**0,181**
Calibration loss	**0,018**	**0,025**	**0,014**
Refinement loss	**0,165**	**0,176**	**0,167**

Brier score met 38 vaste intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	**Disc**	**Tree**
Brier score	**0,174**	**0,196**	**0,167**
Calibration loss	**0,016**	**0,029**	**0,003**
Refinement loss	**0,158**	**0,167**	**0,164**

Brier score met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,167**	**0,178**	**0,166**
Calibration loss	**0,006**	**0,010**	**0,001**
Refinement loss	**0,161**	**0,168**	**0,165**

Brier score met 8 vaste intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,181**	**0,186**	**0,181**
Calibration loss	**0,014**	**0,01**	**0,016**
Refinement loss	**0,167**	**0,176**	**0,165**

Brier score met 11 vaste intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,180**	**0,188**	**0,180**
Calibration loss	**0,012**	**0,009**	**0,015**
Refinement loss	**0,168**	**0,179**	**0,165**

Brier score met 38 vaste intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,168**	**0,178**	**0,167**
Calibration loss	**0,006**	**0,009**	**0,002**
Refinement loss	**0,162**	**0,169**	**0,165**

Bijlage 2: Resultaten van de 9 bestanden gebruikmakend van de minimum refinement loss

Databestand 1

Brier score met 9 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,263	0,331	0,212
Calibration loss	0,057	0,094	0,032
Refinement loss	0,206	0,237	0,180

Brier score met 17 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,225	0,270	0,222
Calibration loss	0,027	0,046	0,050
Refinement loss	0,198	0,224	0,172

Brier score met 36 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,201	0,251	0,169
Calibration loss	0,020	0,037	0,004
Refinement loss	0,181	0,214	0,165

Brier score met 9 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,224**	**0,227**	**0,175**
Calibration loss	**0,033**	**0,033**	**0,005**
Refinement loss	**0,191**	**0,194**	**0,170**

Brier score met 17 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,186**	**0,203**	**0,171**
Calibration loss	**0,007**	**0,019**	**0,001**
Refinement loss	**0,179**	**0,184**	**0,170**

Brier score met 36 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,185**	**0,201**	**0,168**
Calibration loss	**0,008**	**0,019**	**0,003**
Refinement loss	**0,177**	**0,182**	**0,165**

Databestand 2

Brier score met 9 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,251**	**0,315**	**0,196**
Calibration loss	**0,061**	**0,084**	**0,036**
Refinement loss	**0,190**	**0,231**	**0,160**

Brier score met 15 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,206	0,251	0,174
Calibration loss	0,029	0,041	0,026
Refinement loss	0,177	0,210	0,148

Brier score met 35 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,162	0,194	0,152
Calibration loss	0,012	0,029	0,005
Refinement loss	0,150	0,165	0,147

Brier score met 9 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,200	0,196	0,164
Calibration loss	0,023	0,023	0,002
Refinement loss	0,177	0,173	0,162

Brier score met 15 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,176	0,175	0,156
Calibration loss	0,015	0,009	0,005
Refinement loss	0,161	0,166	0,151

Brier score met 35 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,156**	**0,166**	**0,152**
Calibration loss	**0,005**	**0,005**	**0,004**
Refinement loss	**0,151**	**0,161**	**0,148**

Databestand 3

Brier score met 9 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,249**	**0,305**	**0,234**
Calibration loss	**0,053**	**0,074**	**0,050**
Refinement loss	**0,196**	**0,231**	**0,184**

Brier score met 13 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,230**	**0,276**	**0,221**
Calibration loss	**0,038**	**0,051**	**0,042**
Refinement loss	**0,192**	**0,225**	**0,179**

Brier score met 29 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,184**	**0,210**	**0,186**
Calibration loss	**0,011**	**0,020**	**0,008**
Refinement loss	**0,173**	**0,190**	**0,178**

Brier score met 9 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,223**	**0,216**	**0,192**
Calibration loss	**0,037**	**0,020**	**0,008**
Refinement loss	**0,186**	**0,196**	**0,184**

Brier score met 13 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,210**	**0,205**	**0,188**
Calibration loss	**0,028**	**0,016**	**0,004**
Refinement loss	**0,182**	**0,189**	**0,184**

Brier score met 29 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,179**	**0,192**	**0,184**
Calibration loss	**0,006**	**0,006**	**0,002**
Refinement loss	**0,173**	**0,186**	**0,182**

Databestand 4

Brier score met 13 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,232**	**0,293**	**0,201**
Calibration loss	**0,048**	**0,066**	**0,027**
Refinement loss	**0,184**	**0,227**	**0,174**

Brier score met 16 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,197	0,244	0,167
Calibration loss	0,021	0,042	0,001
Refinement loss	0,176	0,202	0,166

Brier score met 33 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,180	0,214	0,168
Calibration loss	0,012	0,028	0,002
Refinement loss	0,168	0,186	0,166

Brier score met 13 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,199	0,200	0,176
Calibration loss	0,026	0,025	0,006
Refinement loss	0,173	0,175	0,170

Brier score met 16 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,185	0,194	0,168
Calibration loss	0,015	0,020	0,002
Refinement loss	0,170	0,174	0,166

Brier score met 33 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,174**	**0,180**	**0,168**
Calibration loss	**0,011**	**0,007**	**0,002**
Refinement loss	**0,163**	**0,173**	**0,166**

Databestand 5

Brier score met 10 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,255**	**0,321**	**0,224**
Calibration loss	**0,057**	**0,085**	**0,043**
Refinement loss	**0,198**	**0,236**	**0,181**

Brier score met 12 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,215**	**0,261**	**0,199**
Calibration loss	**0,033**	**0,041**	**0,034**
Refinement loss	**0,182**	**0,220**	**0,165**

Brier score met 41 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,171**	**0,201**	**0,169**
Calibration loss	**0,016**	**0,026**	**0,005**
Refinement loss	**0,155**	**0,175**	**0,164**

Brier score met 10 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,255**	**0,215**	**0,183**
Calibration loss	**0,061**	**0,028**	**0,010**
Refinement loss	**0,194**	**0,187**	**0,173**

Brier score met 12 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,214**	**0,191**	**0,174**
Calibration loss	**0,039**	**0,015**	**0,003**
Refinement loss	**0,175**	**0,176**	**0,171**

Brier score met 41 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,165**	**0,178**	**0,167**
Calibration loss	**0,007**	**0,008**	**0,002**
Refinement loss	**0,158**	**0,170**	**0,165**

Databestand 6

Brier score met 10 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,258**	**0,324**	**0,219**
Calibration loss	**0,058**	**0,091**	**0,039**
Refinement loss	**0,200**	**0,233**	**0,180**

Brier score met 13 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,210	0,247	0,190
Calibration loss	0,022	0,035	0,026
Refinement loss	0,188	0,212	0,164

Brier score met 36 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,187	0,225	0,167
Calibration loss	0,012	0,028	0,003
Refinement loss	0,175	0,197	0,164

Brier score met 10 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,234	0,216	0,177
Calibration loss	0,040	0,028	0,007
Refinement loss	0,194	0,188	0,170

Brier score met 13 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,200	0,189	0,171
Calibration loss	0,019	0,008	0,001
Refinement loss	0,181	0,181	0,170

Brier score met 36 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,175**	**0,188**	**0,165**
Calibration loss	**0,008**	**0,013**	**0,001**
Refinement loss	**0,167**	**0,175**	**0,164**

Databestand 7

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,242**	**0,221**	**0,227**
Calibration loss	**0,054**	**0,034**	**0,042**
Refinement loss	**0,188**	**0,187**	**0,185**

Brier score met 16 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,204**	**0,260**	**0,201**
Calibration loss	**0,023**	**0,041**	**0,036**
Refinement loss	**0,181**	**0,219**	**0,165**

Brier score met 37 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,179**	**0,211**	**0,168**
Calibration loss	**0,013**	**0,032**	**0,003**
Refinement loss	**0,166**	**0,179**	**0,165**

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,203**	**0,221**	**0,176**
Calibration loss	**0,016**	**0,034**	**0,006**
Refinement loss	**0,187**	**0,187**	**0,170**

Brier score met 16 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,184**	**0,197**	**0,170**
Calibration loss	**0,012**	**0,018**	**0,002**
Refinement loss	**0,172**	**0,179**	**0,168**

Brier score met 37 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,174**	**0,187**	**0,167**
Calibration loss	**0,007**	**0,009**	**0,002**
Refinement loss	**0,167**	**0,178**	**0,165**

Databestand 9

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,280**	**0,358**	**0,244**
Calibration loss	**0,075**	**0,117**	**0,035**
Refinement loss	**0,205**	**0,241**	**0,209**

Brier score met 17 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,216	0,255	0,209
Calibration loss	0,024	0,038	0,029
Refinement loss	0,192	0,217	0,180

Brier score met 35 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,215	0,255	0,208
Calibration loss	0,026	0,042	0,029
Refinement loss	0,189	0,213	0,179

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,240	0,239	0,190
Calibration loss	0,045	0,046	0,008
Refinement loss	0,195	0,193	0,182

Brier score met 17 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,194	0,205	0,182
Calibration loss	0,009	0,014	0,004
Refinement loss	0,185	0,191	0,178

Brier score met 35 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,194	0,203	0,182
Calibration loss	0,009	0,014	0,004
Refinement loss	0,185	0,189	0,178

Databestand 10

Brier score met 9 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,261	0,336	0,234
Calibration loss	0,063	0,100	0,029
Refinement loss	0,198	0,236	0,205

Brier score met 12 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,225	0,279	0,209
Calibration loss	0,036	0,055	0,037
Refinement loss	0,189	0,224	0,172

Brier score met 36 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,175	0,200	0,167
Calibration loss	0,011	0,028	0,003
Refinement loss	0,164	0,172	0,164

Brier score met 9 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,212**	**0,229**	**0,183**
Calibration loss	**0,030**	**0,040**	**0,006**
Refinement loss	**0,182**	**0,189**	**0,177**

Brier score met 12 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,192**	**0,202**	**0,175**
Calibration loss	**0,019**	**0,028**	**0,004**
Refinement loss	**0,173**	**0,174**	**0,171**

Brier score met 36 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,167**	**0,179**	**0,166**
Calibration loss	**0,004**	**0,010**	**0,001**
Refinement loss	**0,163**	**0,169**	**0,165**

Bijlage 3 : Resultaten van de 9 bestanden gebruikmakend van de 3 gecalibreerde probabilteiten

Databestand 1

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,195	0,238	0,169
Calibration loss	0,009	0,028	0,002
Refinement loss	0,186	0,210	0,167

Brier score met 20 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,186	0,210	0,167
Calibration loss	0,008	0,018	0,001
Refinement loss	0,178	0,192	0,166

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,187	0,195	0,167
Calibration loss	0,009	0,010	0,002
Refinement loss	0,178	0,185	0,165

Brier score met 20 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,182**	**0,193**	**0,168**
Calibration loss	**0,005**	**0,008**	**0,003**
Refinement loss	**0,177**	**0,185**	**0,165**

Databestand 2

Brier score met 14 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,168**	**0,213**	**0,153**
Calibration loss	**0,007**	**0,022**	**0,005**
Refinement loss	**0,161**	**0,191**	**0,148**

Brier score met 23 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,162**	**0,191**	**0,152**
Calibration loss	**0,013**	**0,026**	**0,004**
Refinement loss	**0,149**	**0,165**	**0,148**

Brier score met 14 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,158**	**0,164**	**0,152**
Calibration loss	**0,003**	**0,003**	**0,004**
Refinement loss	**0,155**	**0,161**	**0,148**

Brier score met 23 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,156	0,165	0,152
Calibration loss	0,006	0,004	0,004
Refinement loss	0,150	0,161	0,148

Databestand 3

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,195	0,242	0,188
Calibration loss	0,013	0,028	0,009
Refinement loss	0,182	0,214	0,179

Brier score met 19 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,186	0,225	0,187
Calibration loss	0,009	0,018	0,009
Refinement loss	0,177	0,207	0,178

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,183	0,194	0,183
Calibration loss	0,005	0,006	0,001
Refinement loss	0,178	0,188	0,182

Brier score met 19 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,180**	**0,191**	**0,184**
Calibration loss	**0,003**	**0,005**	**0,002**
Refinement loss	**0,177**	**0,186**	**0,182**

Databestand 4

Brier score met 10 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,205**	**0,251**	**0,188**
Calibration loss	**0,028**	**0,038**	**0,022**
Refinement loss	**0,177**	**0,213**	**0,166**

Brier score met 22 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,174**	**0,201**	**0,168**
Calibration loss	**0,008**	**0,024**	**0,002**
Refinement loss	**0,166**	**0,177**	**0,166**

Brier score met 10 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,186**	**0,184**	**0,173**
Calibration loss	**0,021**	**0,011**	**0,003**
Refinement loss	**0,165**	**0,173**	**0,170**

Brier score met 22 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,167	0,175	0,169
Calibration loss	0,006	0,004	0,003
Refinement loss	0,161	0,171	0,166

Databestand 5

Brier score met 15 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,177	0,224	0,171
Calibration loss	0,012	0,022	0,006
Refinement loss	0,165	0,202	0,165

Brier score met 21 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,173	0,212	0,171
Calibration loss	0,013	0,023	0,006
Refinement loss	0,160	0,189	0,165

Brier score met 15 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,166	0,179	0,174
Calibration loss	0,006	0,007	0,003
Refinement loss	0,160	0,172	0,171

Brier score met 21 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,165	0,180	0,167
Calibration loss	0,006	0,008	0,002
Refinement loss	0,159	0,172	0,165

Databestand 6

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,183	0,213	0,169
Calibration loss	0,008	0,016	0,005
Refinement loss	0,175	0,197	0,164

Brier score met 19 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,179	0,204	0,168
Calibration loss	0,008	0,017	0,004
Refinement loss	0,171	0,187	0,164

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,179	0,177	0,166
Calibration loss	0,008	0,003	0,002
Refinement loss	0,171	0,174	0,164

Brier score met 19 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,175**	**0,178**	**0,165**
Calibration loss	**0,004**	**0,007**	**0,001**
Refinement loss	**0,171**	**0,171**	**0,164**

Databestand 7

Brier score met 14 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,183**	**0,219**	**0,172**
Calibration loss	**0,011**	**0,025**	**0,007**
Refinement loss	**0,172**	**0,194**	**0,165**

Brier score met 21 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	**0,179**	**0,214**	**0,168**
Calibration loss	**0,011**	**0,022**	**0,003**
Refinement loss	**0,168**	**0,192**	**0,165**

Brier score met 14 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	**0,178**	**0,185**	**0,167**
Calibration loss	**0,010**	**0,007**	**0,002**
Refinement loss	**0,168**	**0,178**	**0,165**

Brier score met 21 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,174	0,184	0,167
Calibration loss	0,007	0,007	0,002
Refinement loss	0,167	0,177	0,165

Databestand 9

Brier score met 12 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,200	0,233	0,188
Calibration loss	0,010	0,024	0,008
Refinement loss	0,190	0,209	0,180

Brier score met 20 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,192	0,215	0,185
Calibration loss	0,007	0,015	0,006
Refinement loss	0,185	0,200	0,179

Brier score met 12 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,188	0,196	0,181
Calibration loss	0,006	0,008	0,003
Refinement loss	0,182	0,188	0,178

Brier score met 20 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,185	0,193	0,181
Calibration loss	0,006	0,007	0,003
Refinement loss	0,179	0,186	0,178

Databestand 10

Brier score met 11 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,175	0,209	0,167
Calibration loss	0,010	0,021	0,003
Refinement loss	0,165	0,188	0,164

Brier score met 23 gemeenschappelijke intervallen en toegepast op gegevens waarop geen calibratietechniek is uitgevoerd

	Log	Disc	Tree
Brier score	0,174	0,198	0,167
Calibration loss	0,011	0,023	0,003
Refinement loss	0,163	0,175	0,164

Brier score met 11 gemeenschappelijke intervallen en met calibratie

	Log	Disc	Tree
Brier score	0,170	0,185	0,166
Calibration loss	0,005	0,007	0,001
Refinement loss	0,165	0,178	0,165

Brier score met 23 gemeenschappelijke intervallen en met calibratie

	Log	**Disc**	**Tree**
Brier score	**0,168**	**0,177**	**0,166**
Calibration loss	**0,004**	**0,006**	**0,001**
Refinement loss	**0,164**	**0,171**	**0,165**