Data Science en Machine Learning: concreet voorbeeld verzekeringsportefeuille

Data Science en Machine Learning zijn hip. Talloze seminars over deze onderwerpen zijn georganiseerd binnen het actuariële werkveld en velen staan er nog op de agenda. Data Science behelst het gehele spectrum van dataverwerking, Machine Learning is een collectie van algoritmes die gebruikt worden voor predictie of classificatie.

Vooralsnog is de aandacht vaak nog vrij globaal: er wordt weinig of niet ingegaan op de technieken en de daadwerkelijke toepassing in de context van verzekeringen. In dit artikel licht ik na een algemene inleiding één specifieke Machine Learningtechniek (Extreme Gradient Boosting) toe, en vergelijk de resultaten ervan, toegepast op een verzekeringsportefeuille, met het in het actuariële werkveld bekende lineaire model.

Dr. H.J. Plat AAG RBA is eigenaar van Richard Plat Consultancy en geeft in die hoedanigheid advies aan verzekeraars en pensioenfondsen op het gebied van waardering en risicomanagement.

1. ACHTERGROND

De reden voor de aandacht voor deze technieken is dat het binnen het actuariële werkveld zowel als bedreiging en als kans wordt gezien. Een bedreiging omdat Data Science een rol kan spelen in een eventuele verdergaande automatisering van het actuariële takenpakket. Een kans omdat actuarissen een rol kunnen spelen in het toepassen van de technieken binnen verzekeraars (met een focus die breder is dan alleen de verzekeringsverplichtingen). Immers, actuarissen zijn al sinds jaar en dag bezig met predictie op basis van data, echter over het algemeen met minder beschikbare observaties dan wat veelal beschikbaar is binnen Machine Learning toepassingen[1]. Evenwel, als actuarissen die rol willen pakken, zullen zij zich moeten bekwamen in de benodigde technieken.

2. MACHINE LEARNING TECHNIEKEN

Er zijn een veelheid van Machine Learning technieken beschikbaar, en de ontwikkeling van nieuwe technieken gaat gestaag verder. Bekende Machine Learning technieken zijn:

- Regressie:
  - Lineair;
  - Logistisch;
  - Ridge / Lasso;
- Decision Tree gebaseerde methodes:
  - Decision Tree;
  - Random Forests;
  - Gradient Boosting;
  - Extreme Gradient Boosting (XGBoost);
- Neurale Netwerken.

Regressie wordt ook als Machine Learning techniek gezien, maar lineaire en logistische regressie zijn technieken die al jarenlang door actuarissen gebruikt worden. Ridge en Lasso regressie zijn varianten waarbij restricties op de parameters worden gehanteerd om multicollineariteit (hoge correlatie tussen verklarende variabelen) te adresseren.

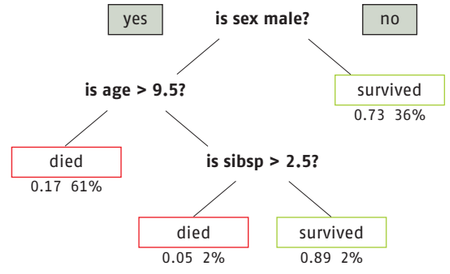
Neurale netwerken zijn gebaseerd op de werking van de hersenen en zenuwen: het zijn zeer uitgebreide en onderling verbonden netwerken van kunstmatige neuronen. Zo'n netwerk bestaat uit verschillende lagen van neuronen, waarbij de neuronen in de verschillende lagen met elkaar gelinkt zijn middels te kalibreren gewichten. Neurale netwerken zijn al tientallen jaren geleden ontwikkeld, maar grote vooruitgang werd meer recent geboekt met onder andere het toevoegen van meerdere lagen in de modellen en het samenvoegen en bewerken van groepen neuronen (convolutionele neurale netwerken).

De decision tree gebaseerde methodes worden in de volgende paragraaf toegelicht. Naast de genoemde methodes is er nog een veelheid aan andere methodes beschikbaar. Ook is het mogelijk om verschillende technieken samen te voegen ('ensembling').

3. DECISION TREE GEBASEERDE METHODES

3.1 Decision tree

Een decision tree model is te vergelijken met de aloude beslisboom. De niveaus en bladeren van de boom worden bepaald door de data steeds verder te splitsen op basis van de verklarende variabelen in de dataset en het minimaliseren van een te kiezen maatstaf zoals de variantie. Dat proces eindigt als verder splitsen geen waarde meer toevoegt aan de kwaliteit van de predictie. Zie het voorbeeld in figuur 1, waarbij een decision tree is gemaakt die voorspelt of een passagier op de Titanic de ramp heeft overleefd of niet op basis van enkele kenmerken ('sibsp' is het aantal gezinsleden aan boord). De getallen onder de bladeren zijn de overlevingskans en het percentage van observaties in de bladeren.



Figuur 1: voorbeeld simpele beslisboom (bron: wikipedia)

Een voordeel van het model is dat het begrijpelijk is en de resultaten goed te interpreteren zijn. Nadelen zijn echter dat de kwaliteit van het model meestal niet goed is en het model over het algemeen ook niet robuust is, de variantie van de schatting is relatief hoog. Het is echter wel een goede bouwsteen gebleken voor onderstaande, beter presterende, modellen.

3.2 Random Forest

De basis voor de Random Forest methode is de 'bootstrap aggregating' (bagging) techniek: hierbij worden door middel van trekking (met teruglegging) *n* sub-datasets uit de data geselecteerd, op basis waarvan *n* decision trees worden gekalibreerd. De predictie voor een nieuwe observatie is vervolgens het gemiddelde van de predictie van de n verschillende decision trees voor deze observatie. Omdat de n decision trees veel op elkaar kunnen lijken als één of een beperkt aantal verklarende variabelen dominant zijn, wordt daarnaast bij iedere splitsing een subset van de verklarende variabelen gebruikt op basis van trekking (met teruglegging).

Random Forests presteren over het algemeen beter dan individuele decision trees en de variantie van de schatting is lager. Het model is echter minder eenvoudig te interpreteren vergeleken met individuele decision trees, al is het wel mogelijk om het relatieve belang van de verschillende verklarende variabelen te rangschikken.

3.3 Gradient Boosting

Gradient Boosting lijkt enigszins op de Random Forest techniek, in de zin dat het gebaseerd is op een aantal decision trees. Echter, bij Gradient Boosting worden de decision trees sequentieel gekalibreerd: na de 1e gekalibreerde decision tree wordt de 2e decision tree gefit aan (bijvoorbeeld) de residuen, de 3e decision tree aan de residuen die resteren na de eerste 2 decision trees, etc. Hierbij is een aantal vrijheidsgraden, zoals het aantal decision trees, aantal niveaus, de te optimaleren functie (bijvoorbeeld minimaliseren absolute fout of gekwadrateerde fout) en het gewicht dat meegegeven wordt aan de nieuwe decision tree (de 'learning rate'). Merk op dat de technieken die gebruikt worden bij Random Forest (bagging en subsets van variabelen) hier desgewenst nog aan toegevoegd kunnen worden.

3.4 Extreme Gradient Boosting (XGBoost)

XGBoost, een afkorting van Extreme Gradient Boosting, is een specifieke implementatie van Gradient Boosting, met een aantal verbeteringen die het model sneller en meer accuraat maakt dan traditionele Gradient Boosting[2]. Deze implementatie is veelvuldig gebruikt in de winnende oplossingen van Data Science competities en is onder andere beschikbaar in de gratis software programma's R en Python. Om deze reden gebruiken we deze techniek voor het uitwerken van het voorbeeld in dit artikel.

4. DATA

Voor het voorbeeld in dit artikel gebruiken we data van de website Kaggle, waarop Data Science competities georganiseerd worden. De data in dit voorbeeld betreft een autoverzekering portefeuille van de Amerikaanse verzekeraar Allstate (zie https://www.kaggle.com/c/allstate-claims-severity). De data set bevat voor 18.318 observaties 116 categorische en 14 continue (geanonimiseerde) verklarende variabelen en het bijbehorende schadebedrag. Het doel is om per observatie het schadebedrag zo goed mogelijk in te schatten middels een model.

5. AANPAK

In dit artikel vergelijken we de resultaten van een standaard lineair model met de resultaten van XGBoost. Daartoe worden voor beide modellen de volgende stappen uitgevoerd (in Python):

a) Allereerst worden de categorische variabelen omgezet in indicator variabelen (0 of 1). Dit verhoogd het aantal verklarende variabelen tot 1153.

b) Vervolgens wordt de dataset willekeurig gesplitst in 10 delen. Het model wordt gefit op 9 delen en de gemiddelde absolute fout wordt gemeten voor het 10e deel. Dit wordt 10 maal herhaald.

c) Het gemiddelde en standaarddeviatie van de 10 resultaten uit stap b) wordt bepaald.

De procedure in stap b) en c) heet cross-validatie en geeft een goed beeld van de performance van het model voor observaties waarop het model niet gefit is. Dit is een methode om het risico op 'overfitting' te mitigeren.

Zoals genoemd in paragraaf 3.3 is er een aantal vrijheidsgraden in het XGBoost model. Daarom worden voor het XGBoost tevens de volgende stappen uitgevoerd:

d) Net als bij Random Forests kan het relatieve belang van verklarende variabelen gerangschikt worden. Eerst is een kalibratie gedaan met alle 1153 verklarende variabelen en vervolgens zijn de verklarende variabelen die geen impact hadden op de uitkomst van het model verwijderd. Dit verlaagd het aantal verklarende variabelen tot 134 en maakt de kalibratie in de volgende stap eenvoudiger.

e) Vervolgens zijn stap b) en c) uitgevoerd voor een grid aan waarden voor het aantal decision trees, het aantal niveaus en de learning rate. Hieruit zijn de 2 beste modellen geselecteerd.

6. RESULTATEN

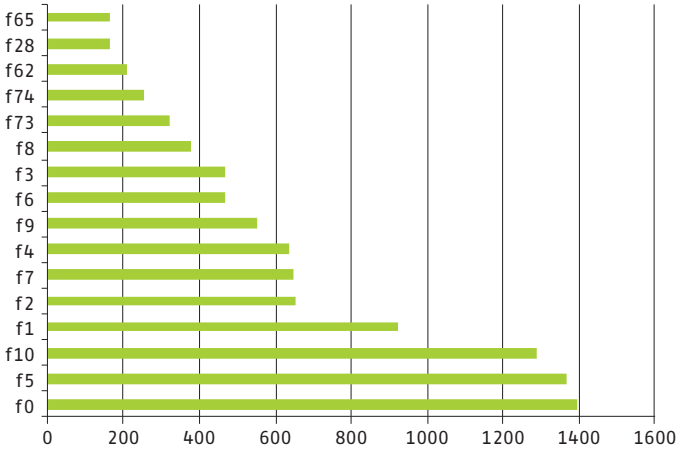
De resultaten zijn weergegeven in tabel 1. XGBoost 1 en 2 zijn de 2 best presterende varianten uit stap e) van de voorgaande paragraaf, XGBoost 3 is een variant op XGBoost 2 waarbij bij iedere splitsing subsets van verklarende variabelen worden gebruikt op basis van trekking (met teruglegging).

Model	Absolute fout		Verschil gemiddelde tov lineair model
	gemiddelde	standaard-deviatie	
Lineair model	1,301	10,1	
XGBoost 1	1,195	17,0	-8,1%
XGBoost 2	1,202	9,9	-7,6%
XGBoost 3	1,202	8,4	-7,6%

Tabel 1: resultaten verschillende modellen

De tabel laat zien dat de absolute fout van de XGBoost modellen 7% à 8% lager is dan een lineair model, waarbij dit voor XGBoost 2 en 3 gepaard gaat met een lagere standaarddeviatie. Deze modellen zijn derhalve te prefereren op basis van sec fit kwaliteit, maar het verschil in kwaliteit is niet groot in dit voorbeeld.

Het relatieve belang van de verklarende variabelen kan voor deze modellen ook worden weergegeven, op basis van de impact van de permutatie van individuele variabelen op de model predictie. Een voorbeeld hiervan is gegeven in figuur 2, waarin het relatieve belang van de 16 meest belangrijke variabelen van XGBoost 2 weergegeven zijn. Merk op dat de namen van de verklarende variabelen geanonimiseerd zijn door Allstate.



Figuur 1: voorbeeld relatieve belang van verklarende variabelen

Uit deze figuur blijkt dat voor het XGBoost 2 model het relatieve belang van de variabelen f0, f5 en f10 het hoogst is. Dit verhoogd de inzichtelijkheid in het model.

7. CONCLUSIE VOORBEELD

In dit artikel is een voorbeeld uitgewerkt van de Machine Learning techniek XGBoost en de resultaten zijn vergeleken met een standaard lineair model. De absolute fout voor de meest optimale XGBoost modellen is 7% a 8% lager dan voor een lineair model, waarbij dit gepaard gaat met een lagere standaarddeviatie. Deze modellen zijn derhalve te prefereren op basis van sec fit kwaliteit. Echter, het verschil in absolute fout is relatief klein. Met het oog op eenvoud van het lineair model en het gegeven dat een betrouwbaarheidsinterval eenvoudiger en sneller te bepalen is, zou een lineair model in dit voorbeeld te prefereren zijn in de praktijk.

8. OVERIGE TOEPASSINGEN

In dit artikel is een voorbeeld uitgewerkt voor het schatten van het schadebedrag, op basis van data die vrij beschikbaar is. Andere toepassingen van Machine Learning technieken zijn bijvoorbeeld:

- Fraudedetectie;
- Pricing;
- Marketing / klantgroep analyse;
- Analyse rijgedrag (op basis van 'telematics' data).

Het grote aantal beschikbare technieken, de mogelijkheid tot samenvoegen van technieken en de doorgaande ontwikkeling van nieuwe technieken maakt in ieder geval dat de technische mogelijkheden groot zijn, ook voor actuarissen! Alles begint echter met voldoende en kwalitatief goede data. ■

1 – Het bedrijf Digital Reasoning heeft bijvoorbeeld een model gekalibreerd met 160 miljard (!) parameters.

2 – Zie 'XGBoost: A Scalable Tree Boosting System' (2016) van Chen en Guestrin.