Centraal Planbureau

Werkloosheidsramingen met machine learning: kan het nog beter?

Het CPB gaat nieuwe tijdreeksmodellen gebruiken om de werkloosheidsramingen te ondersteunen.

De onderzochte machine-learning-methoden voorspellen de werkloosheid beter dan het huidig ondersteunend model.

Onze bevinding dat machine learning nuttig kan zijn om tijdreeksen te voorspellen, komt overeen met die van de economische literatuur.

CPB Achtergronddocument

Bas Scheer

september 2019

Samenvatting

Het CPB gaat nieuwe tijdreeksmodellen gebruiken om de werkloosheidsramingen te ondersteunen. Het CPB heeft hiervoor al eerder een tijdreeksmodel ontwikkeld, het Bayesiaans vector-autoregressief (BVAR) model (Adema, et al., 2018). Dat model voorspelt de werkloosheid beter dan het brede macro-economische model 'Saffier' tot een jaar vooruit. Dit jaar hebben we onderzocht of we de werkloosheidsvoorspellingen nóg verder kunnen verbeteren met behulp van *machine learning*. Het bijzondere aan *machine learning* is dat we geen structuur opleggen aan de modellen. In plaats daarvan zoekt een algoritme patronen in een dataset en maakt op basis daarvan voorspellingen.

De onderzochte *machine learning*-methoden voorspellen de werkloosheid beter dan het huidig ondersteunend model. Hierbij hebben we gebruikgemaakt van twee methoden: *Support Vector Regression* (SVR) en *Random Forest* (RF). Vooral op de termijn tussen één en twee jaar vooruit zijn de voorspelfouten duidelijk kleiner dan die van het huidig ondersteunend model (BVAR). Gemiddeld zitten de SVR en RF er 0,7%-punt naast bij voorspellingen twee jaar vooruit, terwijl de BVAR er gemiddeld 1,2%-punt naast zit. Bij voorspellingen minder ver vooruit presteren de tijdreeksmodellen vergelijkbaar: de gemiddelde fout bij voorspellingen één jaar vooruit is 0,3 tot 0,4%-punt.

Om de werkloosheid te voorspellen gebruiken we zowel administratieve gegevens als vertrouwensindicatoren. Onder vertrouwensindicatoren vallen het consumentenvertrouwen, producentenvertrouwen en de vacature-indicator. Deze indicatoren zijn gebaseerd op de verwachtingen die consumenten en producenten hebben over de economische ontwikkeling, dat maakt ze mogelijke voorspellers van het werkloosheidspercentage. De administratieve gegevens bestaan uit het aantal faillissementen en de ontwikkeling van het aantal WW'ers. Deze geven informatie over het aantal werkzoekenden dat we de komende tijd kunnen verwachten. Alle data zijn openbaar en worden maandelijks gepubliceerd.

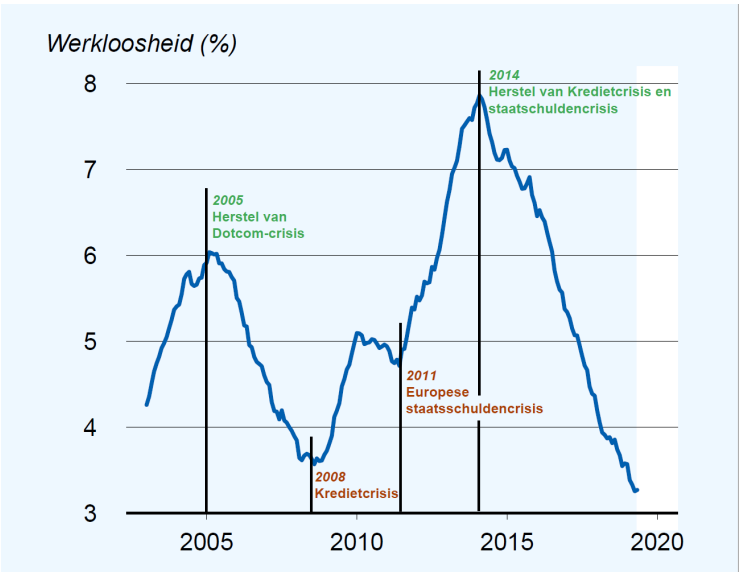
Onze bevinding dat *machine learning* nuttig kan zijn om tijdreeksen te voorspellen, komt overeen met die van de economische literatuur. Uit eerder onderzoek met internationale data blijkt dat deze methoden economische indicatoren, zoals inflatie en werkloosheid, vaak goed kunnen voorspellen (bijvoorbeeld Stasinakis, Sermpinis, Theofilatos en Karathanasopoulos, 2016; Xu, Li, Cheng en Zheng, 2013). Wel geldt dat de kwaliteit van deze voorspellingen sterk afhangt van de hoeveelheid data waar het model op getraind is, en de validatie-methode (Benítez & Bergmeir, 2012).

***Machine learning* kan de bestaande macro-economische modellen niet vervangen, maar kan wel helpen bij het verbeteren van de ramingen.** De tijdreeksmodellen helpen het ramen van de werkloosheid met nauwkeurigere voorspellingen. Zij kunnen de macro-economische modellen van het CPB echter niet vervangen, omdat zij niet voor andere doelen dan voorspellen ingezet kunnen worden. Zo is het niet mogelijk om de BVAR en *machine learning*-modellen te gebruiken om de effecten van beleid mee te nemen in onze raming. Andere (macro-economische) modellen van het CPB, zoals Saffier, zijn daar wel geschikt voor. Bovendien raamt Saffier, naast de werkloosheid, nog veel andere macro-economische variabelen en levert daarmee een consistent beeld op van de economie als geheel.

1 Introductie

Het werkloosheidspercentage is een belangrijke arbeidsmarkindicator in de ramingen van het CPB. Vier keer per jaar wordt deze indicator geraamd voor het lopende en aankomende jaar. Uit eerder onderzoek naar de trefzekerheid van onze ramingen blijkt dat het verschil tussen het geraamde en gerealiseerde werkloosheidspercentage aanzienlijk kan zijn (Adema et al. 2018). Ramingen zijn vooral lastig te maken rond een omslagpunt: wanneer het werkloosheidspercentage van een dalende trend overgaat naar een stijgende trend of andersom. Voorbeelden hiervan zijn de kredietcrisis van 2008, toen de werkloosheid van dalend naar stijgend omsloeg, en de herstelperiode die in 2014 is ingezet, toen de werkloosheid weer begon te dalen na een jarenlange stijging (zie figuur 1.1).

Figuur 1.1 Omslagpunten in het werkloosheidspercentage sinds 2003



Het doel van dit onderzoek is om een model te ontwikkelen dat de kortetermijnwerkloosheidsraming (tot 24 maanden vooruit) kan verbeteren. In voorgaand onderzoek hebben we een aantal methoden ontwikkeld om de werkloosheidsraming te ondersteunen. Hiermee kunnen we betere voorspellingen maken dan het CPB-macromodel tot ongeveer een jaar vooruit, ook rond omslagpunten (Adema, et al., 2018). Er blijft echter vraag naar een ondersteunend model dat ook verder dan een jaar vooruit nauwkeurig voorspelt, omdat deze ramingen gebruikt worden voor het Centraal Economische Plan (CEP) en de Macro Economische Verkenning (MEV). Deze ramingen spelen een belangrijke rol in de sociaal-economische politieke besluitvorming in Nederland. In deze publicaties rapporteren we een projectie van het werkloosheidspercentage voor het lopende en komende jaar. Om het komende jaar te ramen moeten we zowel de rest van het lopende als het volgende jaar projecteren. We ramen dan dus meer dan 12 maanden vooruit (maximaal 24 maanden).

Uit internationaal onderzoek blijkt dat werkloosheid goed kan worden voorspeld met *machine learning*-technieken Dit is een vorm van *data science*, waarbij een algoritme patronen leert herkennen in een dataset. In ons geval willen we het werkloosheidspercentage voorspellen met een set economische indicatoren. Op basis van de patronen die het algoritme vindt tussen het werkloosheidpercentage en economische indicatoren kan het een inschatting maken van het verloop van het werkloosheidspercentage in de nabije toekomst. Verschillende onderzoeken hebben recentelijk aangetoond dat *machine learning* tijdreeksen beter kan

voorspellen dan conventionele econometrische methoden, zoals ARMA-modellen[1] (Xu, Li, Cheng & Zheng, 2013; Plakandaras, Gupta, Gogas & Papadimitriou, 2014; Stasinakis, Sermpinis, Theofilatos & Karathanasopoulos, 2016 en Tyralis & Papacharalampous, 2017). De kwaliteit van deze voorspellingen hangt echter sterk af van de hoeveelheid data waar het model op getraind is en de methode waarmee het model wordt afgesteld (Benítez & Bergmeir, 2012).

Daarom zijn we voor Nederland nagegaan of we met behulp van *machine learning* de voorspelling van de werkloosheid verder kunnen verbeteren. Onze onderzoeksopzet hiervoor wordt besproken in paragraaf 2. Daarin presenteren we de tijdreeksmodellen die we testen, onze methode om de modellen zo goed mogelijk af te stellen en de data waarop we de modellen schatten. De resultaten hiervan bespreken we in paragraaf 3, bestaande uit een overzicht van de voorspelfouten van de modellen. Ten slotte concluderen we en bespreken we mogelijk verder onderzoek in paragraaf 4.

2 Methoden & onderzoekopzet

We onderzoeken twee *machine learning*-modellen waarvan is aangetoond dat ze economische reeksen goed kunnen voorspellen. Deze twee zijn: Support Vector Regression (SVR) en Random Forest (RF). Bijvoorbeeld, Atef en Imbens (2019) hebben dit soort machine learning-methoden recent aangeprezen als mogelijke toevoegingen aan de econometrische gereedschapskist. Mits deze methoden goed worden afgesteld zijn ze uitstekende voorspellers.

Het belangrijkste verschil tussen *machine learning* en traditionele statistiek is dat de eerste direct gericht is op voorspellen, terwijl de tweede meer gericht is op het aantonen van relaties tussen variabelen.[2] Traditionele statistiek is meer gericht op het vinden van empirisch bewijs waarmee een hypothese over de relatie tussen variabelen bevestigd of afgewezen kan worden. Hierbij brengen we vooraf informatie in op basis van economische theorie, zoals de relevante variabelen en de vorm van het verband tussen die variabelen. *Machine learning* zoekt een model dat goed voorspelt binnen een dataset. Daarbij is in eerste instantie onbekend wat de relevante variabelen zijn en welke vorm de verbanden aannemen. De set van voorspellers moet worden samengesteld door de onderzoekers, maar *machine learning* pikt de meest relevante variabelen en verbanden op. Dit soort methoden presteren relatief goed als er veel voorspellers zijn (t.o.v. het aantal observaties) die veel op elkaar lijken (sterk gecorreleerd zijn).

Voor ons doel – het voorspellen van de werkloosheid – is het belangrijk dat een model goed voorspelt als het nieuwe data krijgt. Dat is immers hoe het in de praktijk wordt ingezet: er komt elke maand nieuwe informatie beschikbaar en op basis daarvan maken we een nieuwe voorspelling. Om ervoor te zorgen dat de modellen in de praktijk goed voorspellen gebruiken we een afstelmethode die de juiste mate van complexiteit van het model vindt. Een model met maar enkele verklarende variabelen is waarschijnlijk niet complex genoeg en loopt het risico om relevante verbanden in de data te missen. Dit wordt *underfitting* genoemd. Een complex model met te veel verklarende variabelen loopt echter het risico om verbanden te vinden die alléén bestaan in de trainingsdata en niet in de nieuwe data. Hierdoor voorspelt een te complex model slecht op nieuwe data. In jargon heet dit *overfitting*. Met onze afstelmethode proberen we een balans te vinden tussen *underfitting* en *overfitting*, zodat het model zo klein mogelijke fouten maakt als het voorspelt op basis van nieuwe data.

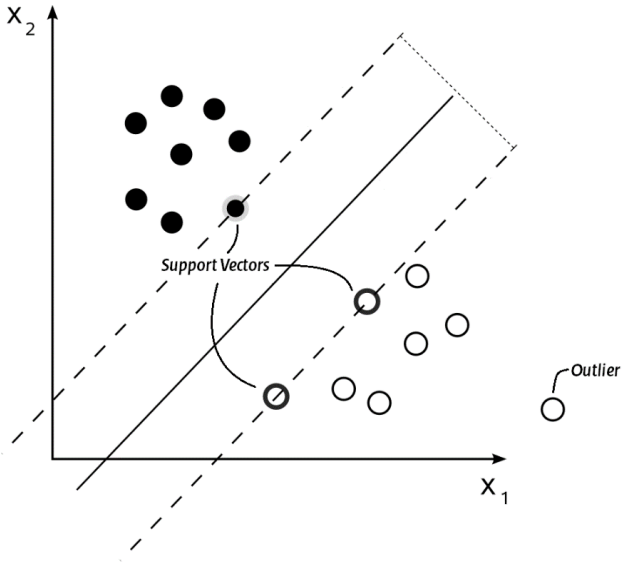
[1] Autoregressive Moving Average (ARMA) model
[2] Bestaande uit geschatte parameters met een onzekerheidsinterval daaromheen

2.1 Methode 1: Support Vector Machines

***Support Vector Machines (SVMs)* zijn veelbelovende voorspelmodellen.** Oorspronkelijk is dit algoritme ontwikkeld om classificatieproblemen op te lossen. Je kunt er bijvoorbeeld spamfilters mee maken: het scheidt binnenkomende mail dan in twee klassen, gewenste mail en ongewenste mail. Bij *Support Vector Regression* (SVR) is deze techniek aangepast, zodat het kan omgaan met continue variabelen als uitkomstvariabele, zoals het werkloosheidspercentage. Het SVM- of SVR-algoritme zoekt een functie waarmee de voorspelde waarden zo dicht mogelijk bij de daadwerkelijke waarden liggen. Bij een classificatieprobleem betekent dit dat zoveel mogelijk observaties juist ingedeeld worden, bij een regressieprobleem dat de waarde die uit de functie komt zo dicht mogelijk bij de geobserveerde waarde ligt.

Het SVR-algoritme werkt goed als voorspelmodel, omdat het robuust is tegen *outlier*s (observaties die ver van de overige data verwijderd liggen). Daarnaast kan dit algoritme ingewikkelde (niet-lineaire) verbanden schatten. Het idee achter het SVR-algoritme is om een regressielijn te vinden die zo goed mogelijk op de data past. In dat opzicht lijkt deze methode op een standaard lineaire regressie (OLS).[3] Een belangrijk verschil is dat bij een SVR een specifieke set observaties bepalen hoe de regressielijn loopt, terwijl bij OLS alle observaties meetellen. De regressielijn van de SVR wordt namelijk gebaseerd op de observaties die dicht bij deze regressielijn liggen. Deze observaties 'dragen' de regressielijn: als één van de observaties verschuift, dan verandert ook de regressielijn. Daarom worden deze observaties ook wel '*support vectors*' genoemd. Figuur 2.1 illustreert een SVM waarin de variabelen X1 en X2 gebruikt worden om de uitkomst - zwart of witte stip - te classificeren. De zwarte, doorgetrokken regressielijn wordt zo gekozen dat de afstand tussen de *support vectors* en de regressielijn gemaximaliseerd wordt. In dit voorbeeld zijn er drie *support vectors*, namelijk de zwarte en de twee witte cirkels die op de gestreepte lijnen liggen.

Figuur 2.1 Support Vectors



Een SVR is minder gevoelig voor *outliers* dan OLS-schattingen. Dit komt omdat de OLS-schattingen afhangen van alle observaties (inclusief outliers), terwijl de SVR-schattingen gebaseerd zijn op de dichtstbijzijnde observaties (de *support vectors*). Het algoritme zoekt de regressielijn die het beste op de data past en vindt in dit proces de *support vectors*.[4] Bij het bepalen van de regressielijn kan het SVR-algoritme makkelijk

[3] OLS staat voor Ordinary Least Squares.
[4] Zie voor de mathematische afleiding hiervan bijvoorbeeld Smola & Schölkopf (2004).

niet-lineaire effecten inbouwen, waardoor de regressielijn allerlei vormen aan kan nemen. Zie voor meer uitleg bijvoorbeeld Vapnik (1998), Schölkopf & Smola (2001) en Athey & Imbens (2019). Een nadeel van het SVR-algoritme, en *machine learning*-methoden in het algemeen, is dat de uitkomsten moeilijker te interpreteren zijn. OLS levert een set lineaire verbanden tussen de voorspellers en de uitkomstvariabele met een betrouwbaarheidsinterval. Daarmee kan eenvoudig het effect van elk afzonderlijke voorspeller in kaart kan worden gebracht. Een SVR-schatting geeft niet zulke makkelijk te interpreteren resultaten. Om te bepalen hoe belangrijk elke voorspeller is voor de geschatte uitkomstvariabele moet een extra gevoeligheidsanalyse worden uitgevoerd.[5] Hieronder volgt een technische uitleg van de toepassing van het SVR-algoritme en een overzicht van de belangrijkste parameters.

De SVR heeft drie hyperparameters die we moeten afstellen om goede voorspellingen te kunnen maken. Hyperparameters zijn parameters die we niet schatten op basis van de data, maar zelf moeten vaststellen. Elk van deze parameters bepaalt op zijn eigen manier de balans tussen *over-* en *underfitting*. Daarom zoeken we naar een goede combinatie van deze parameters. De beste combinatie vinden we met een datagedreven afstelmethode. Deze afstelmethode wordt verder toegelicht in subsectie 2.3. De eerste hyperparameter is v ligt tussen 0 en 1 en bepaalt welk aandeel van de data wordt gezien als *outlier* en daardoor wordt genegeerd in de regressie (Schölkopf B. , Smola, Williamson, & Bartlett, 2000). De tweede is de *Cost*-parameter C. Deze weegt de eenvoud van het model af ten opzichte van de *fit* op de data. De derde is γ. Deze geeft de gewenste mate van niet-lineariteit van het model aan. Daarin nemen we drie mogelijke waarden voor γ mee, gebaseerd op een op de dataset geschatte bandbreedte (Caputo, Sim, Furesjo, & Smola, 2002). Voor C nemen we zes mogelijke waarden mee.[6] En v zetten we gelijk aan een standaardwaarde van 0.5.[7] Het aantal mogelijke combinaties van hyperparameters komt hiermee op 18 (3 x 6 x 1).

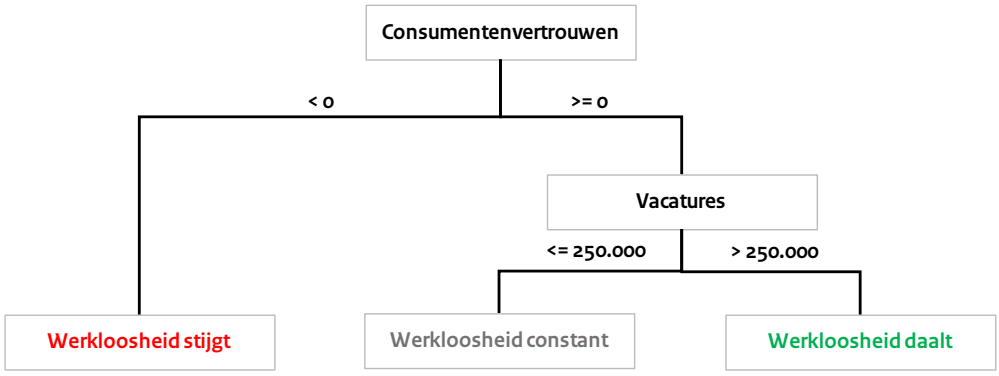
2.2 Methode 2: Random Forests

Ook *Random Forest (RF)* is een veelbelovende methode voor het maken van voorspelmodellen. De *Random forest* (RF) is een veelgebruikte en effectief methode om flexibele regressies te schatten (Athey & Imbens, 2019). Een deel van hun populariteit hebben ze te danken aan het feit dat ze relatief eenvoudig te gebruiken zijn: over het algemeen hoeft een RF niet uitvoerig te worden afgesteld voordat het goede voorspellingen oplevert. Daarnaast is de RF goed in het bepalen welke variabelen relevant zijn en kunnen ze goed omgaan met non-lineaire verbanden en interacties tussen variabelen.

Het RF-algoritme werkt goed als voorspelmodel, omdat het robuust is tegen *outlier*s, ingewikkelde (niet-lineaire) verbanden kan schatten en interacties tussen variabelen kan meenemen. Het idee achter het RF-algoritme is gebaseerd op *regression trees* (regressiebomen). Het basisprincipe hiervan is om de dataset op te splitsen in stukken en op basis daarvan een voorspelling te doen. Stijgt het werkloosheidspercentage bijvoorbeeld altijd als het consumentenvertrouwen onder een bepaalde waarde komt? Dan wordt de dataset onderverdeeld op basis van het consumentenvertrouwen. Is het consumentenvertrouwen lager dan de vastgestelde waarde, dan voorspelt het algoritme een stijging van het werkloosheidspercentage (zie figuur 2.2). Daalt het werkloosheidspercentage als het consumentenvertrouwen hoog is, maar alleen als er ook voldoende vacatures zijn? Dan kan de regressieboom een verdere opsplitsing maken op basis van het aantal vacatures (zie figuur 2.2). Zo ontstaat er een beslissingsboom waarmee de ontwikkeling van het werkloosheidspercentage wordt voorspeld. Hieronder volgt een technische uitleg van de toepassing van het RF-algoritme en een overzicht van de belangrijkste parameters die we moeten afstellen.

[5] Zie appendix voor een gedetailleerde uitleg van deze gevoeligheidsanalyse (A.1).
[6] Deze waarden zijn: $2^{-2}, 2^{0}, 2^{2}, 2^{4}, 2^{8}$ *en* 2^{10}
[7] Standaardwaarde in LIBSVM (R-package), Chang & Lin (2011).

Figuur 2.2 Fictief voorbeeld van een regression tree (regressieboom)



Het RF-algoritme voorspelt op basis van beslisbomen die de data herhaaldelijk opsplitsen. In de eerste stap wordt de data opgesplitst in twee stukken. Het algoritme kiest de splitsing - op basis van één van de voorspellers - die de gemiddelde voorspelfout minimaliseert.[8] Vervolgens kunnen deze groepen verder worden opgesplitst, zodat vier groepen ontstaan. Dit gaat door totdat het aantal observaties in een groep kleiner is dan een vooraf bepaalde grenswaarde. Deze groep wordt daarna niet verder opgesplitst. Deze grenswaarde bepaalt daarmee het aantal opsplitsingen van de *regression tree*. Als de grenswaarde te laag is, dan leidt dit tot splitsingen die wél tot betere voorspellingen leiden in de trainingsdata, maar die tot slechtere voorspellingen leiden op nieuwe data (*overfitting*). Als de grenswaarde te hoog is, dan worden er te weinig splitsingen gemaakt en voorspelt de boom slecht op zowel de trainingsdata als op nieuwe data (*underfitting*). [9]

Een *random forest* voorkomt *overfitting* door een groot aantal *regression trees* te combineren. Als er maar één *regression tree* geschat wordt, is de kans op *overfitting* groot (Friedman, Hastie, & Tibshirani, 2001) (Athey & Imbens, 2019). Het RF-algoritme maakt daarom een groot aantal verschillende regressiebomen (*trees*), die samen een bos *(forest)* vormen. De voorspelling van het *RF*-algoritme is de gemiddelde voorspelling van dit bos. Door elke *regression tree* anders op te bouwen pikken ze verschillende verbanden uit de dataset op, waardoor de kans op *overfitting* afneemt. Het algoritme genereert verschillende *regression trees* door iedere boom te baseren op een *gebootstrapte* steekproef uit de trainingsdata.[10] Daarnaast kan ze bij iedere splitsing slechts uit een beperkt aantal variabelen kiezen en worden de variabelen in deze set willekeurig bepaald.[11] Dit levert een groot aantal voorspellingen op per observatie (één voorspelling voor iedere steekproef waarin de observatie voorkomt) en het *random forest* kiest als uiteindelijke voorspelling het gemiddelde hiervan.

De drie belangrijkste hyperparameters van een *random forest* zijn het aantal bomen dat geschat wordt, het aantal variabelen dat per splitsing overwogen wordt en de grenswaarde die bepaalt hoeveel splitsingen er gemaakt worden. Een te klein aantal bomen vergroot het risico op *overfitting*. Een te hoog aantal bomen heeft weinig nadelige gevolgen op de voorspelkwaliteit, maar vergroot wel de rekentijd. We zetten het aantal te schatten bomen gelijk aan 500.[12] Daarnaast neemt het RF-algoritme voor elke splitsing een

[8] Binnen iedere splitsing is de voorspelling gelijk aan de gemiddelde waarde van de uitkomstvariabele.
[9] Deze procedure geldt enkel voor *regression trees*, waarbij de afhankelijke variabele continue is. De splitsingscriteria is anders als de te verklaren variabele bivariaat is. In dit geval wordt gesproken van een *classification tree*.
[10] Iedere *gebootstrapte* steekproef wordt gemaakt door observaties met teruglegging te trekken uit de trainingsdata.
[11] Het aantal variabelen waaruit gekozen wordt, is doorgaans gelijk aan een derde van het aantal variabelen in de data (Friedman, Hastie, & Tibshirani, 2001).
[12] Dit aantal bomen wordt bijvoorbeeld ook gebruikt door Liaw & Wiener (2002) in hun voorbeeld van de toepassing van het *Random Forest*-algoritme op een bestaande dataset. In het *R-package* dat zij geschreven hebben voor het schatten van *Random Forests* is dit de standaardwaarde. Daarnaast hebben we de voorspelfout in de validatieset in kaart gebracht bij een verschillend aantal bomen. Deze uitkomsten suggereren dat 500 bomen ruim voldoende is (zie appendix, subsectie A.2).

willekeurige set variabelen uit te de totale set om deze splitsing op te baseren. Het aandeel van de variabelen dat per splitsing uitgekozen wordt is de tweede hyperparameter. Deze zetten we gelijk aan een derde van het aantal variabelen in de trainingset.[13] De grenswaarde voor het minimaal aantal observaties waarop een verdere splitsing gebaseerd moet zijn zetten we gelijk aan de standaardwaarde van 5.[14]

2.3 Afstelmethode

Onze methode om de *machine learning*-modellen af te stellen geeft een realistisch beeld van de voorspelkwaliteit wanneer er nieuwe data beschikbaar komt. Dit doen we door de modellen eerst af te stellen en vervolgens voorspellingen te laten doen op data die ze nog niet gezien hebben. Dat is immers hoe ze in de praktijk worden ingezet. Er komt elke maand nieuwe informatie beschikbaar en op basis daarvan maken ze een voorspelling. Deze voorspellingen zijn *out-of-sample*-voorspellingen: dat wil zeggen dat de data waarmee de voorspelling is gemaakt niet is gebruikt om te bepalen wat de beste parameters zijn. Alle resultaten die we presenteren zijn gebaseerd op deze *out-of-sample*-voorspellingen. Zo zorgen we ervoor dat de voorspelkwaliteit die we in dit onderzoek vinden overeenkomt met de voorspelkwaliteit in de praktijk.

Om onze modellen af te stellen delen we de data op in sets voor trainen, valideren en evalueren. De trainingset gebruiken we om de algoritmes patronen te laten ontdekken. Vervolgens gebruiken we deze schattingen om voorspellingen te doen in de validatieset. Op basis van de voorspelkwaliteit in de validatieset kunnen we bepalen welke afstelling van de hyperparameters het best werkt. Vervolgens bekijken we of de afstelling die we daarmee selecteren wel echt goed werkt door het geselecteerde model nog eens een *out-of-sample* voorspelling te laten doen. Dit noemen we de evaluatieset. Deze set bevat de voorspellingen die het model in een realistische *setting* gemaakt zou hebben. Op basis daarvan illustreren we de voorspelkwaliteit van de verschillende algoritmes.

Om te beginnen krijgt het algoritme de eerste 60 maanden – van januari 2003 tot en met januari 2008 - van de dataset om de beste variant te selecteren. Het eerste deel van deze 60 maanden is echter niet bruikbaar voor de schattingen, omdat we die gebruiken voor het aanmaken van de variabelen in de dataset.[15] Het aantal observaties dat we hierdoor niet kunnen gebruiken is gelijk aan het aantal vertraagde termen (in dit geval 6), plus de voorspelhorizon, plus één. Bij een voorspelhorizon van één maand betekent dit dat we 8 van de 60 maanden niet kunnen gebruiken voor de schattingen. Van de overige 52 maanden wordt de eerste 66,7% gebruikt om het model patronen te laten ontdekken. Dit is de trainingset. Vervolgens maken we op basis hiervan een voorspelling voor de eerstvolgende maand na de trainingset. Deze voorspelling is de eerste van de validatieset. Nu schuiven we de hele dataset één maand naar voren: we voegen een extra observatie toe aan de trainingset en schatten het model opnieuw. Daarna maken we weer een voorspelling voor de maand na de trainingset en voegen die toe aan de validatieset. Dit herhalen we totdat maand 60 (januari 2008) van de dataset is bereikt. De validatieset bevat nu de voorspellingen van alle varianten van het model tussen maand 43 (juli 2006) en 60 (december 2007). In het stroomschema hieronder staat de afstelmethode van de SVR schematisch weergegeven (figuur 2.4).

---

[13] Dit is de standaardwaarde voor het *Random Forest R-package* van Liaw & Wiener. Volgens Breiman en de makers van het R-package is de voorspelkwaliteit van het *Random Forest*-algoritme niet sterk afhankelijk van deze hyperparameter (Liaw & Wiener, 2002 en Breiman, 2001).

[14] Dit is de standaardwaarde voor het *Random Forest R-package* van Liaw & Wiener (2002).

[15] Deze zijn nodig voor het aanmaken van de vertraagde termen, uitrekenen van eerste verschillen en het aanmaken van de vooruitlopende term van de uitkomstvariabele. Het aantal observaties dat we niet mee kunnen nemen is gelijk aan het aantal vertraagde termen van de voorspellers (in dit geval zes maanden), plus één observatie voor het berekenen van de eerste verschillen, plus de voorspelhorizon voor het aanmaken van de vooruitlopende term. Deze niet bruikbare observaties halen we uit de training- en validatieset, om ervoor te zorgen dat de evaluatieset voor de verschillende voorspelhorizons dezelfde periode beslaat. Daardoor zijn de verschillen in voorspelfouten tussen voorspelhorizons met elkaar te vergelijken.

De beste variant van het model wordt gekozen op basis van de voorspelkwaliteit in de validatieset. We kiezen de beste variant als de afstelling van het model met de laagste gemiddelde kwadratische fout. Met de geselecteerde variant maken we vervolgens een *out-of-sample* voorspelling voor de 61ste maand (januari 2008). Deze voorspelling is de eerste in de evaluatieset. Nu schuiven we de hele dataset weer een maand naar voren: we voegen een extra observatie toe aan de trainingset, maken op basis hiervan een voorspelling in de validatieset en beoordelen weer welke variant de beste is. Op basis van deze variant maken we een *out-of-sample*-voorspelling voor de 62ste maand (februari 2008). Dit herhalen we totdat het einde van de dataset is bereikt (mei 2019). Er zijn dan in totaal 137 voorspellingen in de evaluatieset waarop we de *out-of-sample*-voorspelkwaliteit van de modellen kunnen beoordelen. Omdat er elke maand weer een nieuwe observatie bijkomt in de validatieset, kan de beste afstelling van het model gedurende deze periode van 137 maanden veranderen. Op die manier kan het model zich aanpassen aan structurele veranderingen in de economie en het aantal beschikbare observaties.

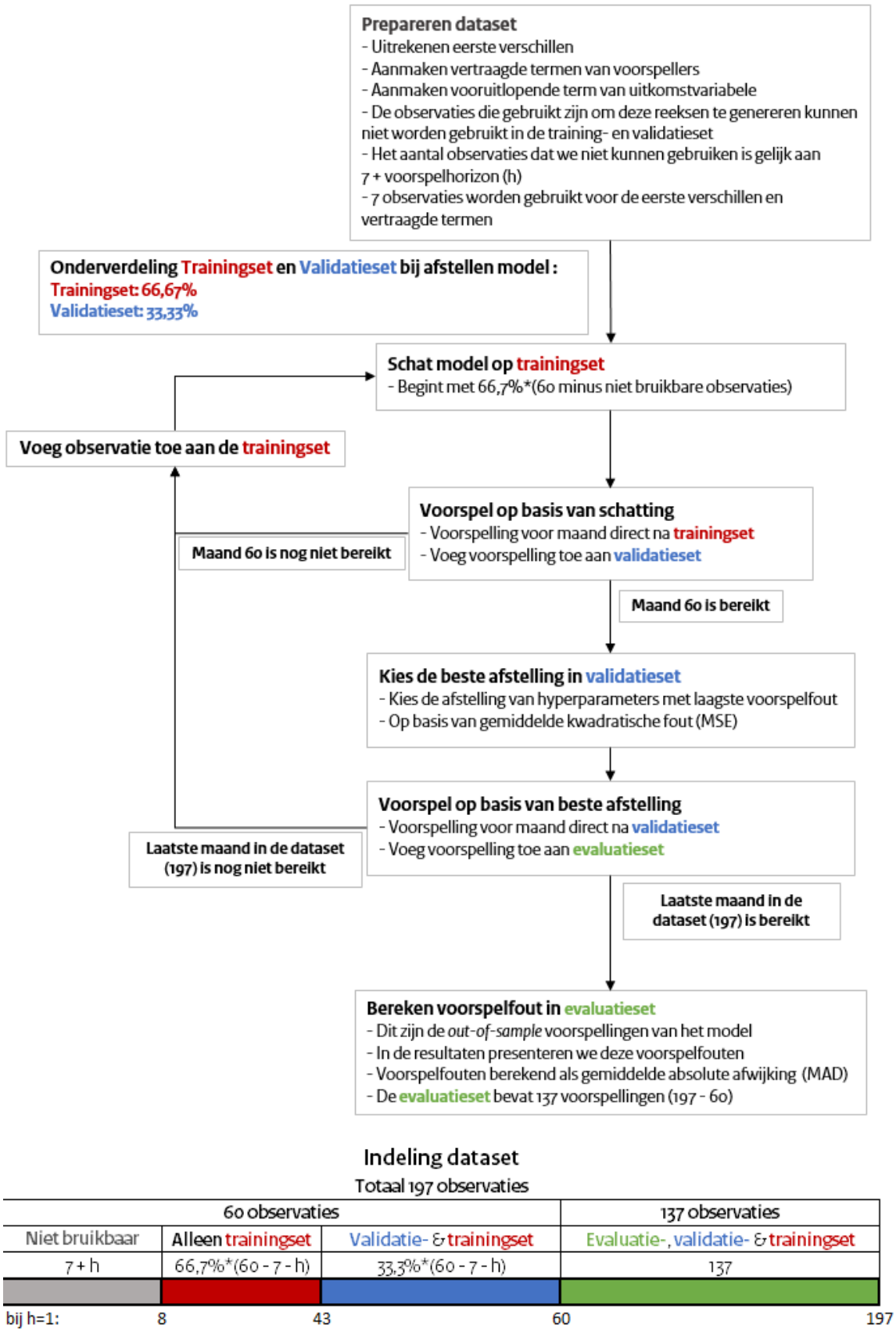
Bij een voorspelhorizon van langer dan één maand vooruit gebruiken we precies dezelfde afstelmethode. Het enige verschil is dat het aantal maanden dat we niet kunnen gebruiken groter is. Dit aantal is namelijk gelijk aan $6 + h + 1$, waarbij h de voorspelhorizon is (in maanden). Hierdoor is het aantal observaties in de training- en validatieset kleiner. Het aantal observaties in de evaluatieset houden we even groot, zodat de verschillen in voorspelfout tussen voorspelhorizons goed met elkaar te vergelijken zijn.

De afstelmethode van het RF is vergelijkbaar met die voor de SVR. Dit doen we – net als bij de SVR – door de data op te delen in drie sets: training, validatie en evaluatie. De trainingset en validatieset samen beginnen met de eerste vijf jaar (60 maanden) van de dataset. De verdeling tussen trainingset en validatieset is 62,3% om 37,7%.[16] Net als bij de SVR wordt op basis van de validatieset gekeken welke afstelling goed voorspelt.[17] Met de gekozen afstelling maken we de voorspelling voor de eerstvolgende maand. Deze voorspelling is de eerste van de evaluatieset. Vervolgens schuift de hele dataset één maand naar voren en worden de trainingset, validatieset en evaluatieset uitgebreid. Dit herhalen we totdat we de laatste maand in de dataset bereiken (mei 2019). De evaluatieset bevat nu de voorspellingen die het model realistisch zou hebben gemaakt tussen juli 2008 (de 61ste maand) en mei 2019 (de 197ste maand).

[16] Het standaardaandeel voor de trainingsset is 63,2% in het *R-package randomForest* van Liaw & Wiener (2002). In onze dataset betekent dit, bij de eerste iteratie van de trainingsmethode, 38 maanden.

[17] We hebben de voorspelfout in de validatieset in kaart gebracht bij een verschillend aantal bomen. Deze uitkomsten suggereren dat 500 bomen ruim voldoende is (zie appendix, subsectie A.2).

Figuur 2.4 – Stroomschema van de afstelmethode



2.4 Benchmarks

De voorspelkwaliteit van de *machine learning*-modellen vergelijken we met het huidige ondersteunend model. Het huidige ondersteunende model is een Bayesiaans vector-autoregressief model (BVAR). Dit model is ontwikkeld naar aanleiding van eerder onderzoek van het CPB waaruit bleek dat dit model betere voorspellingen maakt dan het CPB-macromodel (Saffier) tot ongeveer een jaar vooruit. VAR-modellen hebben snel last van *overfitting*, omdat het complexe modellen zijn met veel parameters. Een Bayesiaans VAR-model gebruikt extra informatie (zogenaamde 'priors') om *overfitting* tegen te gaan. Bijvoorbeeld de Minnesota-prior, waarin wordt verondersteld dat de data een autoregressief proces[18] volgt: het pad van elke variabele in de dataset wordt alleen bepaald door zijn eigen historie. Als uit schattingen van het model blijkt dat deze prior de data niet goed beschrijft, dan laat het VAR-model ook verbanden tussen variabelen toe. Door de prior past het model zich dus niet meteen volledig aan de verbanden in de dataset, waardoor de kans op *overfitting* kleiner is. Uit eerder onderzoek blijkt een combinatie van verschillende priors[19] het best te werken voor het voorspellen van het werkloosheidspercentage. Zie voor deze resultaten en meer uitleg over de BVAR het achtergronddocument over werkloosheidsramingen (Adema, et al., 2018) en het technisch achtergronddocument over BVAR-modellen (De Wind, 2015).

We vergelijken de machine learning-modellen ook met een naïeve voorspelregel. Deze voorspelregel is simpelweg dat de werkloosheid niet verandert.[20] In juli 2019 is het werkloosheidspercentage bijvoorbeeld 3,3%, dus is de voorspelling op basis van de ze regel dat het werkloosheidspercentage na juli 2019 3,3% blijft. Deze vuistregel geeft de bovengrens van de voorspelfouten aan: een model moet minimaal betere voorspellingen doen dan deze simpele regel.

Om als ondersteunend model ingezet te worden dienen de *machine learning*-modellen minstens zo goed te voorspellen als deze twee *benchmarks*.

2.5 Data

Om de werkloosheid te voorspellen gebruiken we zowel vooruitkijkende vertrouwensindicatoren als administratieve gegevens. Onder de vertrouwensindicatoren vallen het consumentenvertrouwen, producentenvertrouwen en de vacature-indicator. Deze indicatoren zijn gebaseerd op de verwachtingen die consumenten en producenten hebben van de economische ontwikkeling. Dat maakt ze mogelijke voorspellers van het werkloosheidspercentage. De administratieve gegevens bestaan uit het aantal faillissementen en de ontwikkeling van het aantal WW'ers. Deze geven informatie over het aantal werkzoekenden dat we de komende tijd kunnen verwachten. De WW-data wordt maandelijks gepubliceerd door het UWV. De overige gegevens worden maandelijks gerapporteerd door het CBS. Zie tabel 2.1 voor een overzicht van de totale dataset. Onze dataset bestaat uit 198 observaties van deze variabelen en loopt van januari 2003 (eerste consistente waarneming) tot en met mei 2019 (laatste waarneming).

De modellen gebruiken tot zes maanden aan vertraagde termen in hun voorspelling. De SVR en RF worden geschat op één vertraagde term in niveaus en zes vertraagde termen in eerste verschillen.[21] De

[18] Om precies te zijn een *AR*(1)-proces.
[19] De combinatie-prior (Sims, 1998) combineert drie verschillende mogelijkheden. De eerste component volgt de Minnesota-prior. De tweede prior veronderstelt dat er geen coïntegratie tussen de reeksen in het model is. De derde prior heet de single-unit-root-prior en laat de mogelijkheid van coïntegratie in het model wel toe. De combination-prior is hierdoor consistent met een aantal verschillende typen eenvoudige VAR-modellen.
[20] In statistiekjargon: een *Random Walk.*
[21] Zie appendix voor specificaties van de modellen (subsectie A.3).

uitkomstvariabele van de modellen is de verandering in het werkloosheidspercentage over de voorspelhorizon. Stel dat we op basis van de beschikbare data in juni 2019 een voorspelling maken voor een jaar later (juni 2020), dan voorspellen de modellen hoeveel het werkloosheidspercentage tussen die twee perioden gaat veranderen. Door die verandering op te tellen bij het werkloosheidspercentage van juni 2019 krijgen we het voorspelde werkloosheidspercentage in juni 2020. De BVAR gebruikt zes vertraagde termen in niveaus om op te schatten. Het voorspelt direct het niveau van het werkloosheidspercentage. De BVAR wordt is als ondersteunend model op dezelfde manier gespecificeerd.

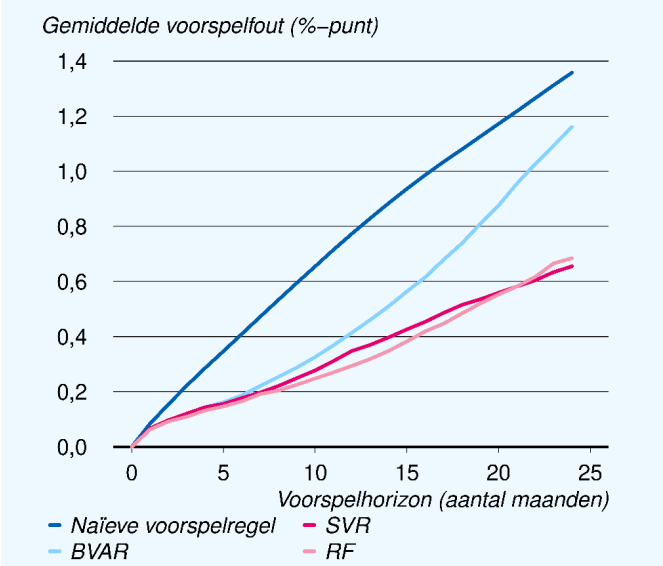
Tabel 2.1 Overzicht van de dataset

Variabelnaam	Omschrijving	Bron
Werkloosheidspercentage	De werkloosheid als percentage van de bevolking 15-75 jaar	CBS
Faillissementen	De procentuele verandering in het aantal faillissementen, gecorrigeerd voor aantal zittingsdagen	CBS
WW-uitkeringen	De procentuele verandering van het aantal WW-gerechtigden	UWV
Instroom WW	Instroom in de WW als percentage van de beroepsbevolking	UWV
Uitstroom WW naar werk	Uitstroom uit de WW omdat werk gevonden is, als percentage van het aantal WW-gerechtigden	UWV
Uitstroom WW tijdslimiet	Uitstroom uit de WW omdat de maximale termijn van de uitkering bereikt is, als percentage van het aantal WW-gerechtigden	UWV
Vacature-indicator	Indicator die aangeeft in welke richting het aantal vacatures in het bedrijfsleven zich naar verwachting zal ontwikkelen. Gemiddelde van vacature-indicatoren van de industrie, de bouwnijverheid en de commerciële dienstverlening. Deze (deel)vacature-indicatoren zijn afgeleid uit verschillende seizoengecorrigeerde variabelen van de maandelijkse conjunctuuronderzoeken: Conjunctuurenquête Nederland (COEN), voorheen de Conjunctuurtest (CT)	CBS
Consumentenvertrouwen	Opvattingen en verwachtingen van Nederlandse consumenten ten aanzien van de algemene economische ontwikkelingen en de eigen financiële situatie. Het gemiddelde van de saldi van de percentages van positieve en negatieve antwoorden op de vragen over de economische situatie in de afgelopen en komende 12 maanden, de financiële situatie van het huishouden in de afgelopen en komende 12 maanden en of het een gunstige tijd is om grote aankopen te doen.	CBS
Producentenvertrouwen	Stemmingsindicator van de ondernemers in de industrie die de richting aan geeft waarin de industriële productie zich naar verwachting zal ontwikkelen. De indicator is het gemiddelde van drie deelindicatoren: verwachte bedrijvigheid, het oordeel over de orderpositie en het oordeel over de voorraden.	CBS

3 Resultaten

De *machine learning*-modellen zijn goede voorspellers van de werkloosheid. Vooral als de voorspelhorizon langer is dan een jaar zijn de voorspelfouten relatief klein ten opzichte van de BVAR (zie figuur 3.1). Gemiddeld zitten de SVR en RF er 0,7%-punt naast bij voorspellingen twee jaar vooruit, terwijl de BVAR er gemiddeld 1,2%-punt naast zit. Bij voorspellingen minder ver vooruit presteren alle tijdreeksmodellen echter vergelijkbaar: de gemiddelde fout bij voorspellingen één jaar vooruit is 0,3 tot 0,4%-punt. Er is geen duidelijk verschil tussen de voorspelfouten van de SVR en RF: ze presteren over de hele voorspelhorizon vergelijkbaar. Alle modellen zijn ruim beter dan de naïeve voorspelregel.

Figuur 3,1 Machine learning-modellen maken kleine voorspelfouten

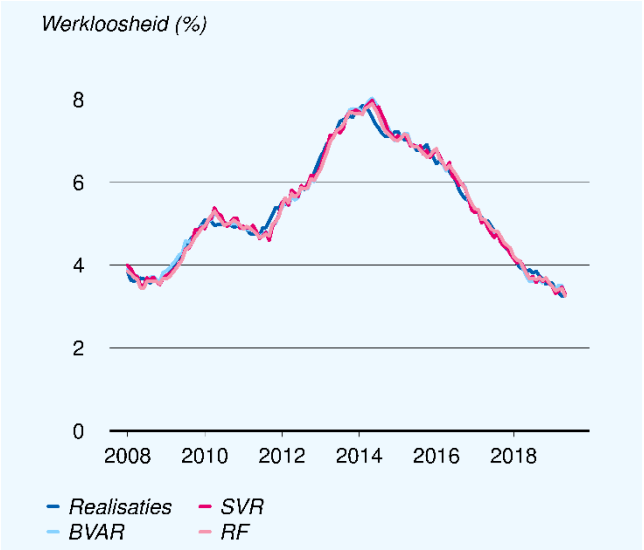


De voorspelfouten zijn groter rond omslagpunten in het werkloosheidspercentage. Vooral bij een langere voorspelhorizon – vanaf 12 maanden - wordt het verschil tussen het voorspelde en daadwerkelijke werkloosheidspercentage groter rond de omslagpunten (Zie figuur 3.2). Dit geldt vooral voor de BVAR aan het begin van de dataset. In de oploop naar de kredietcrisis voorzag dit model nog een te sterke daling van het werkloosheidspercentage, waardoor het rond dit omslagpunt relatief grote voorspelfouten maakte. Na het plaatsvinden van dit omslagpunt voorzag het echter juist een te sterke oploop, waardoor het rond en na de tijdelijke herstelperiode in 2011 te hoge voorspellingen maakte. De SVR en RF laten dit patroon minder sterk zien. De voorspelfouten zijn een stuk kleiner dan die van de BVAR bij deze omslagpunten. Echter, bij het meest recente omslagpunt – het begin van de herstelperiode van 2014 – is dit verschil veel kleiner. Ook daarna, wanneer er geen omslagpunten meer zijn presteren de BVAR, SVR en RF vergelijkbaar. De hogere gemiddelde voorspelfouten van de BVAR ligt dus vooral aan de slechte voorspellingen in de periode 2007-2011.

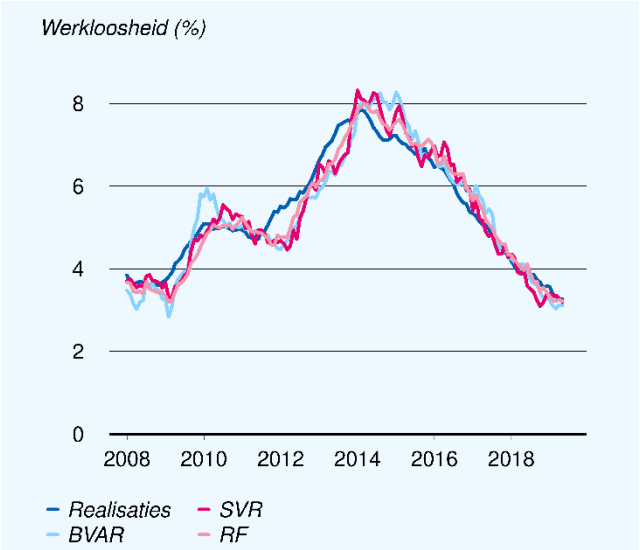
Alle modellen hebben in de recente periode kleinere voorspelfouten gemaakt dan in eerdere perioden. Zoals eerder besproken is dit patroon voor de BVAR het sterkst. Het is onduidelijk wat hiervan de oorzaak is. Een mogelijke verklaring is dat de modellen de patronen in de data beter leren kennen naarmate ze meer maanden hebben kunnen zien, en dat de *machine learning*-modellen dit sneller oppikken dan de BVAR. Een tweede verklaring is dat voorspellen in de tumultueuze crisisperiode van 2008-2012 in het algemeen moeilijker was, maar dat de *machine learning*-modellen hier beter mee om kunnen gaan. Welke van deze twee de grootste rol speelt is lastig uit elkaar te trekken. Daarom zullen we voorspelkwaliteit van de drie modellen moeten blijven volgen over de komende jaren om hier meer inzicht in te krijgen.

Figuur 3.2 Verschil tussen voorspelde en gerealiseerde werkloosheid groter rond omslagpunten

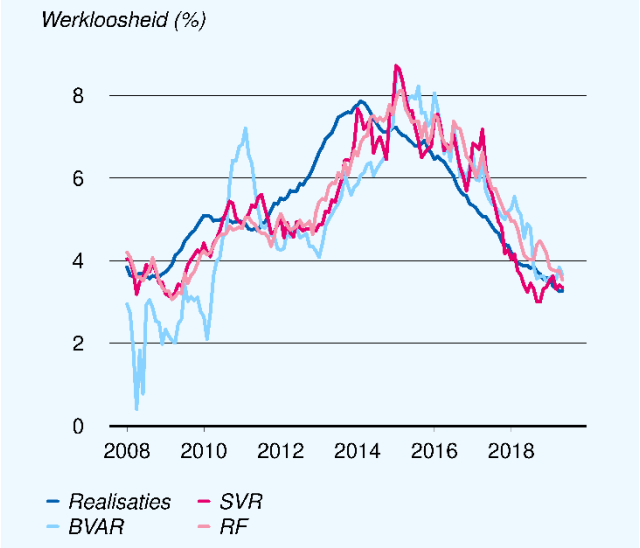
Voorspelling 3 maanden vooruit



Voorspelling 12 maanden vooruit

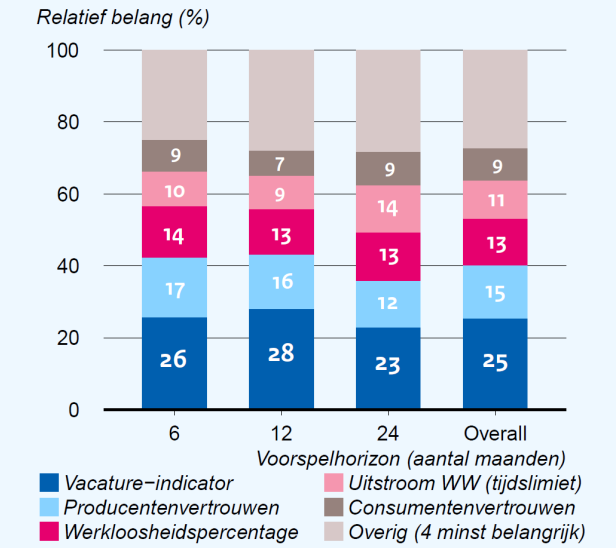


Voorspelling 24 maanden vooruit



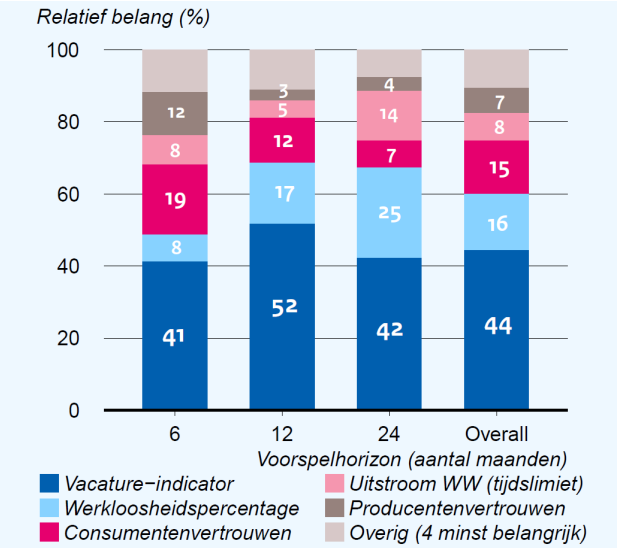
De SVR selecteert de vacature-index, het producentenvertrouwen en het recente verloop van het werkloosheidspercentage als belangrijkste voorspellers. Over het algemeen bepalen deze drie samen 53% van de voorspellingen (zie figuur 3.3). De overige 47% wordt bepaald door de andere 6 voorspellers. Dit beeld is vergelijkbaar als we de voorspelhorizon uitsplitsen in 6, 12 en 24 maanden. De verschillen tussen de korte en lange voorspelhorizon zijn niet groot. Bij de langste voorspelhorizon is het producentenvertrouwen iets minder belangrijk (met 12%) en is de uitstroom vanuit de WW door het bereiken van de uitkeringstermijn belangrijker (met 14%).

Figuur 3.3 Belangrijkste voorspellers volgens de SVR



Noot: het relatieve belang van de voorspellers is berekend over de hele steekproef (van januari 2003 t/m mei 2019)

Figuur 3.4 Belangrijkste voorspellers volgens de RF



Noot: het relatieve belang van de voorspellers is berekend over de hele steekproef (van januari 2003 t/m mei 2019).

De voorspellers die de RF als belangrijkste selecteert, komen grotendeels overeen met de selectie van de SVR. Ook volgens de RF is belangrijkste voorspeller de vacature-indicator. Bovendien bestaat de top vijf uit dezelfde voorspellers, al verschilt de rangorde van plaats twee t/m vijf (zie figuur 3.4). Er zijn echter ook wat verschillen zichtbaar. Het eerste wat opvalt is dat het RF-algoritme een duidelijkere selectie maakt: de top drie samen bepalen 75% van de voorspelling. Dit komt vooral doordat de vacature-indicator als belangrijkste indicator een groter aandeel is toegewezen, met 44% bij de RF versus 25% bij de SVR. De rest van de voorspellers spelen dus een kleinere rol dan bij de SVR. Blijkbaar zijn de splitsingen gemaakt op basis van de vacature-indicator beduidend informatiever dan van de overige voorspellers. Daarnaast zijn de verschillen tussen de korte en lange voorspelhorizon groter. Bij de voorspelling op kortere termijn (6 maanden) is het consumentenvertrouwen relatief belangrijker (met 19%), terwijl bij voorspellingen op wat langere termijn (24 maanden) de vertraagde termen van het werkloosheidspercentage zelf belangrijker zijn (met 25%). De vacature-indicator is op elke voorspelhorizon het meest belangrijk – altijd meer dan 40% – met als piek meer dan de helft (52%) bij een voorspelhorizon van 12 maanden.

4 Conclusie & verder onderzoek

Het CPB gaat de SVR en RF gebruiken als ondersteunende modellen, als aanvulling op de BVAR. Voor de raming van het lopend jaar – tot maximaal 12 maanden vooruit – lopen de resultaten van deze drie modellen niet ver uiteen. Op basis van onze onderzoeksresultaten verwachten we een gemiddelde voorspelfout van 0,3-0,4%-punt voor deze modellen bij een voorspelhorizon van 12 maanden. Bij ramingen langer dan 12 maanden vooruit zal het CPB ook alle drie de modellen inzetten. Daarbij houden we wel rekening met het feit dat de BVAR gemiddeld minder goed heeft voorspeld op deze horizon. Bij een voorspelhorizon van 24 maanden verwachten we een gemiddelde voorspelfout van rond de 0,7%-punt. Dit betreft wel een gemiddelde, rond omslagpunten zijn de voorspelfouten groter.

Een belangrijke uitbreiding voor de *machine learning*-modellen is om voor elke werkloosheidsraming apart aan te kunnen geven wat het effect is van elke afzonderlijke voorspeller op de voorspelde waarde. Dit is nuttige informatie voor het duiden van de werkloosheidsraming. Zo kunnen we bijvoorbeeld aangeven dat een recente daling in het consumentenvertrouwen heeft geleid tot een hogere werkloosheidsraming. In de huidige analyse hebben we alleen onderzocht welke variabelen over de dataset als geheel belangrijk zijn. Dit uitsplitsen naar individuele maanden is geen simpele opdracht. De modellen zijn ingewikkeld: de verbanden tussen het werkloosheidspercentage en de voorspellers zijn non-lineair en kunnen met elkaar interacteren. Voor het huidig ondersteunend model – de BVAR – hebben we hier al een methode voor ontwikkeld. Hierbij kijken we naar het verschil in de voorspelling door van één van de voorspellers net te doen alsof we de meest recente observatie nog niet hebben. Zo weten we wat het bekend worden van deze observatie voor effect heeft gehad op de voorspelling. Door dit voor elke voorspeller uit te voeren hebben we een inschatting van het partiële effect van de voorspellers. Voor de nieuwe *machine learning*-modellen is deze methode nog in ontwikkeling en willen we extra methoden ontwikkelen om de verbanden goed in kaart te brengen.

Het CPB gaat de onderzochte methoden ook testen voor andere macro-economische variabelen. De komende maanden gaan we inventariseren bij welke ramingen vraag is naar ondersteunende modellen en of *machine learning*-methoden daar een nuttige bijdrage kunnen leveren. Mogelijke kanshebbers hiervoor zijn het bbp en de wereldhandel. Verder onderzoek moet uitwijzen welke methoden het meest geschikt zijn en of we hiermee een nuttig ondersteunend model kunnen maken.

Literatuur

Adema, A., Folmer, K., Van heuvelen, H., Kuijpers, S., Luginbuhl, R., & Scheer, B. (2018). *Voorspellen van de werkloosheid: kan het beter?* Den Haag: CPB.

Athey, S., & Imbens, G. W. (2019). Machine Learning Methods Economists Should Know About. *Annual Review of Economics*.

Benítez, J. M., & Bergmeir, C. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 192-213.

Breiman, L. (2001). Random forests. *Machine learning*, 5-32.

Brown, G. (2010). Ensemble Learning. *Encyclopedia of Machine Learning*, 312-320.

Caputo, B., Sim, K., Furesjo, F., & Smola, A. (2002). Appearance-based object recognition using SVMs: which kernel should I use? *Proc of NIPS workshop on Statistical methods for computational experiments in visual processing and computer vision*.

Chang, C. C., & J, L. C. (2011). LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*.

Cortez, P., & Embrechts, M. J. (2013). Using sensitivity analysis and visualization techniques to open black box data mining models. *Information Sciences*, 1-17.

De Wind, J. (2015). *Technical background document for BVAR models used at CPB.* Den Haag: CPB.

Friedman, J., Hastie, T., & Tibshirani, R. (2001). *The elements of statistical learning: data mining, inference and prediction.* New York: Springer series in statistics.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R news*, 18-22.

Plakandaras, V., Gupta, R., Gogas, P., & Papadimitriou, T. (2014). Forecasting the U.S. real house price index. *Economic Modelling*, 259-267.

Polikar, R. (2012). Ensemble learning. *Ensemble machine learning*, 1-34.

Schölkopf, B., & Smola, A. J. (2001). *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Cambridge: MIT Press.

Schölkopf, B., Smola, A., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. *Neural Computation*, 1207–1245.

Sims, C. A. (1998). Bayesian Methods for Dynamic Multivariate Models. *International Economic Review*, *39*(4), 949-968.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing*, 199-222.

Stasinakis, C., Sermpinis, G., Theofilatos, K., & Karathanasopoulos, A. (2016). Forecasting US Unemployment with Radial Basis Neural Networks, Kalman Filters and Support Vector Regressions. *Computational Economics*, 569-587.

Tyralis, H., & Papacharalampous, G. (2017). Variable Selection in Time Series Forecasting Using Random Forests. *Algorithms*, 1-25.

Vapnik, V. (1998). *Statistical Learning Theory.* New York: Wiley&Sons.

Xu, W., Li, Z., Cheng, C., & Zheng, T. (2013). Data mining for unemployment rate prediction using search engine query data. *Service Oriented Computing and Applications*, 33-42.

Appendix

A.1 Methode voor bepalen belangrijkste voorspellers

Om de belangrijkste voorspellers in kaart te brengen gebruiken we een methode ontwikkeld door Cortez en Embrechts (2013). Ze maken gebruik van gevoeligheidsanalyses om voor verschillende *machine learning*-technieken op een consistente manier uit te rekenen wat de bijdrage is van individuele voorspellers. Wij gebruiken hun eendimensionale gevoeligheidsanalyse (1D-SA). Hierbij wordt de bijdrage van elke voorspeller bepaald door de voorspeller waarin we geïnteresseerd zijn te laten variëren, terwijl de overige voorspellers constant worden gehouden (op de gemiddelde waarde in de dataset). De mate waarin de voorspelde waarde hierdoor veranderd geeft aan hoe gevoelig de voorspelling is voor de voorspeller in kwestie. Op deze manier rekenen we de gevoeligheid uit voor elke individuele voorspeller. Door deze uitkomsten met elkaar te vergelijken kunnen we kwantificeren welke voorspellers belangrijk zijn en welke minder. In deze eendimensionale gevoeligheidsanalyse wordt geen rekening gehouden met interacties tussen voorspellers over tijd. Het geeft een algemeen beeld van de algemene bijdrage van elke voorspeller over de gehele dataset. Het voordeel van deze variant is dat er relatief weinig rekentijd nodig is. Hieronder volgt een technische uitleg van de gevoeligheidsanalyse.

We schatten de *machine learning*-modellen op onze dataset, bestaande uit N observaties van M voorspellers en één uitkomstvariabele y. De voorspelde waarde van deze uitkomstvariabele noemen we $\hat{y}$. Elke maand hebben we een vector van één observatie voor elke voorspeller m. Deze vector noemen we x. Het model neemt als *input* de vector x en geeft als *output* de voorspelde waarde $\hat{y}$. Deze functie noemen we P:

$$(1) \quad \hat{y} = P(x)$$

De gevoeligheidsanalyse neemt één van de voorspellers (x_a) en laat deze variëren van de minimumwaarde tot de maximumwaarde. De overige voorspellers worden constant gehouden op hun gemiddelde. Het aantal niveaus van x_a dat we in de gevoeligheidsanalyse opnemen noemen we L. In onze gevoeligheidsanalyses zetten we $L = 7$. We delen het bereik tussen de minimum- en maximumwaarde op in gelijke delen op basis van L niveaus. Heeft x_a bijvoorbeeld een bereik van [0,1], dan nemen we de niveaus $x_{aj} \in \{0.00, 0.17, 0.33, 0.67, 0.83, 1.00\}$ op in de analyse.

De eendimensionale gevoeligheidsanalyse (1D-SA) gaat langs alle voorspellers: $\{x_a : a \in \{1, \dots, M\}\}$. Voor de te analyseren voorspeller worden L niveaus opgenomen $x_{aj} : j \in \{1, \dots, L\}\}$. De waarden voor de andere voorspellers worden constant gehouden op hun gemiddelde. Voor elk niveau wordt met functie (1) uitgerekend wat de bijbehorende voorspelde waarde is. Dit geeft voor elke voorspeller een set van L voorspelde waarden: $\hat{y}_a = \{\hat{y}_{aj} : j \in \{1, \dots, L\}\}$. Deze sets geven aan hoe gevoelig de voorspelde waarde $\hat{y}$ voor elke voorspeller x_a. Hoe groter de variantie in $\hat{y}_{aj}$, des te groter de gevoeligheid voor x_a. De intuïtie hierachter is dat een belangrijke voorspeller voor grotere veranderingen in de uitkomstvariabele zorgt dan minder belangrijke voorspellers. Deze belangrijkheid wordt uitgedrukt met een gevoeligheidsmaatstaf. Hiervoor gebruiken we de gemiddelde absolute afwijking (*average absolute deviation*: AAD):

$$(2) \quad AAD_a = \sum_{j=1}^{L} \left| \frac{\hat{y}_{aj} - \tilde{y}_a}{L} \right|$$

Waarbij $\tilde{y}_a$ de mediane voorspelde waarde in de set van voorspeller x_a is. Hoe hoger AAD_a, des te groter het effect van voorspeller x_a op de voorspelde waarde, dus des te belangrijker x_a. Het relatieve belang van de voorspeller (r_a) wordt gegeven als:

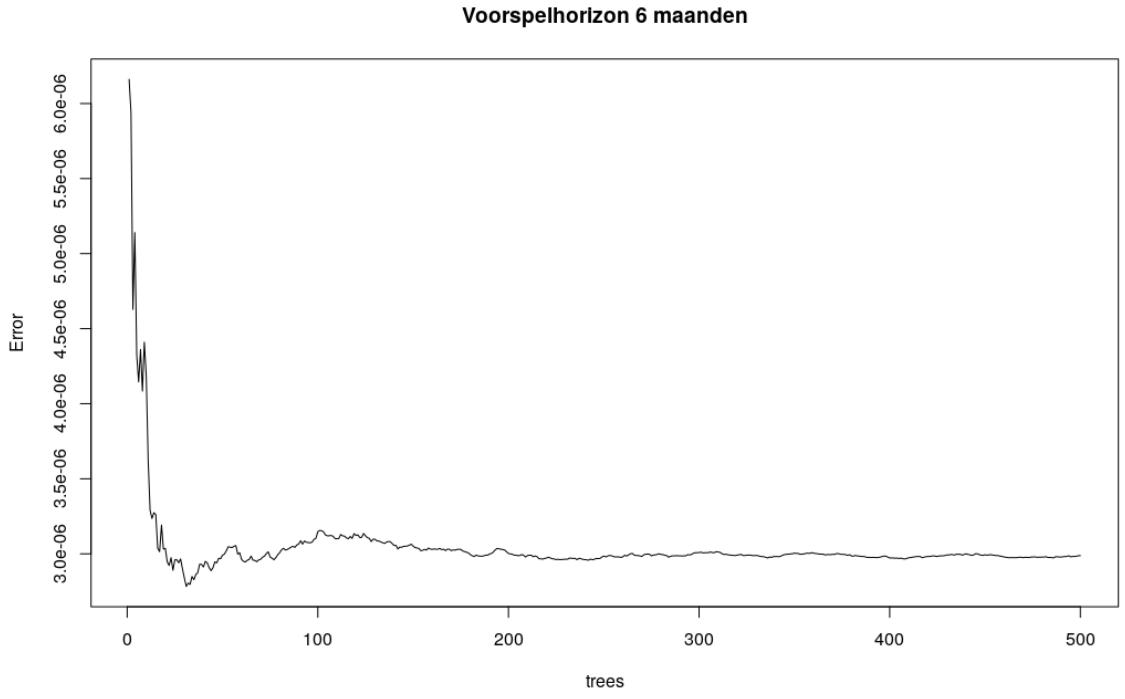
$$(3)\ r_a = \frac{AAD_a}{\sum_{i=1}^{M} AAD_i}$$

In de modellen gebruiken we meerdere vertraagde termen van elke variabele als voorspellers. Met de bovenstaande analyse wordt het relatieve belang van elk van deze vertraagde termen apart uitgerekend. In de resultaten van paragraaf 3 behandelen we de geaggregeerde resultaten: de som van het relatieve belang van alle vertraagde termen. Dit geeft aan hoe belangrijk elk van de 9 voorspellers is in het bepalen van de voorspelling, uitgedrukt in een percentage.

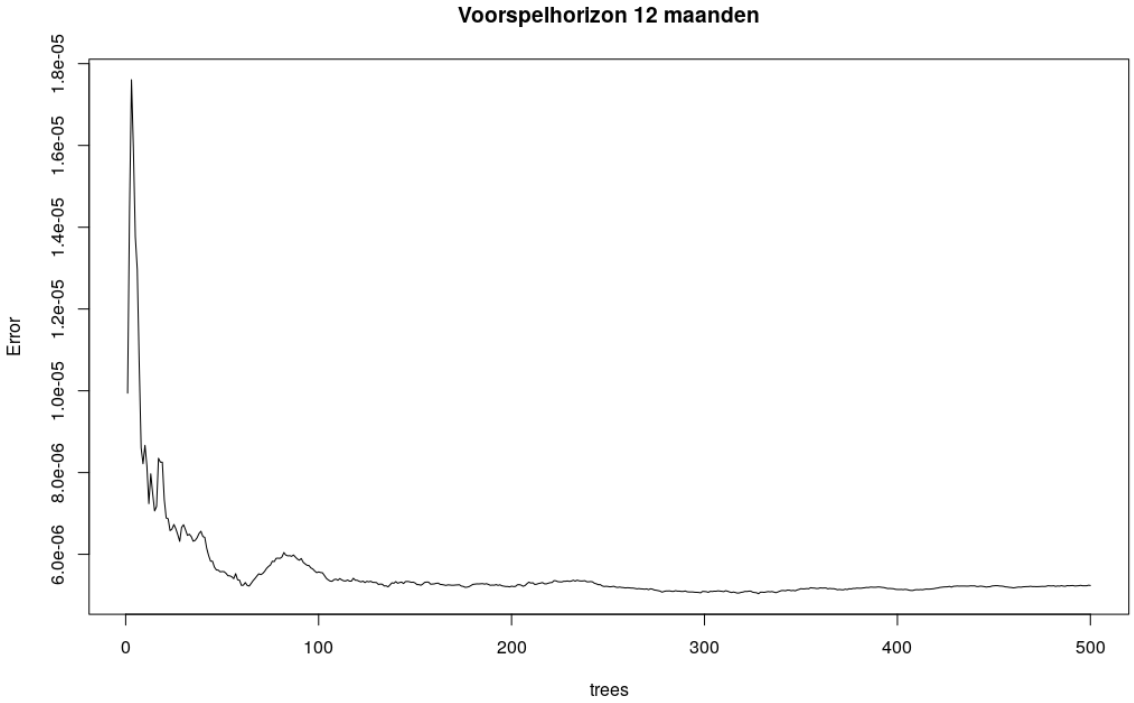
A.2 Aantal regression trees in de Random Forest

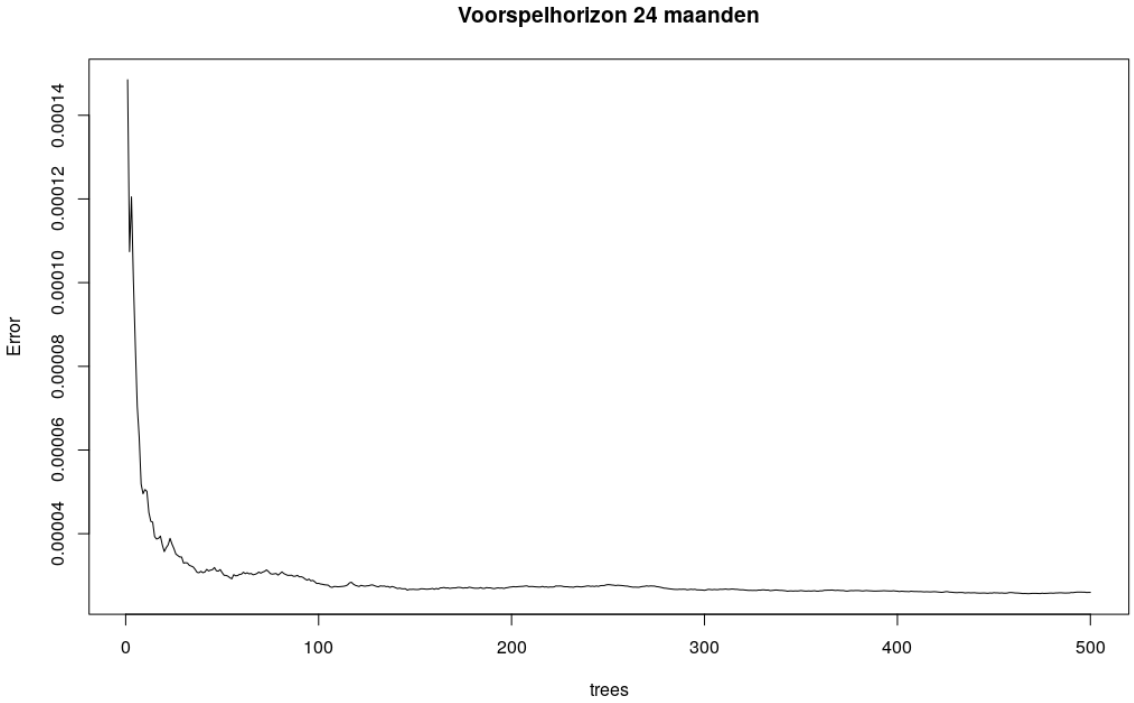
Met het *R-package* voor het schatten van de *Random Forests* hebben we plots gemaakt van de gemiddelde kwadratische voorspelfout in de validatieset (uitkomstvariabele is mutatie in werkloosheidspercentage) op de y-as en het aantal bomen op de x-as. De grafieken van de *Random Forest* geschat over de gehele steekproef (januari 2003 – mei 2019) staan hieronder. Hierin is duidelijk te zien dat na 250 bomen de *error* stabiliseert. Dit is een sterke indicatie dat ons geselecteerd aantal bomen (500) ruim voldoende is.

Aantal bomen versus voorspelfout (error) in validatieset



Aantal bomen versus voorspelfout (error) in validatieset (vervolg)





A.3 Specificaties van de modellen

Specificatie van de SVR & RF:

$$u_{t+h} - u_t = \Delta u_{t+h} = f(X_t) + \varepsilon_t$$

Waarbij:
$u_{t+h} = het\ werkloosheidspercentage\ aan\ het\ eind\ van\ de\ voorspelhorizon\ (t+h)$
$u_{t+h} = het\ werkloosheidspercentage\ in\ de\ maand\ van\ de\ voorspelling\ (t)$
$\Delta u_{t+h} = de\ mutatie\ in\ werkloosheidspercentage\ over\ de\ voorspelhorizon$
$X_t = vector\ van\ vertraagde\ termen\ van\ voorspellers\ in\ de\ maand\ van\ de\ voorspelling\ (t)$
$f(X_t) = functie\ geschat\ met\ RF\ of\ SVR$
$\varepsilon_t = voorspelfout\ in\ mutatie\ van\ het\ werkloosheidspercentage$

X_t bevat van alle voorspellers het niveau van de meest recente maand en zes vertraagde termen in eerste verschillen:

$$X_t = (V_{t-1}, \Delta V_{t-1}, \Delta V_{t-1}, \Delta V_{t-2}, \Delta V_{t-3}, \Delta V_{t-4}, \Delta V_{t-5}, \Delta V_{t-6})$$

$V = vector\ van\ de\ 9\ voorspellers$

De voorspelde waarde voor de mutatie in werkloosheidspercentage van maand t tot maand $t+h$ is:

$$\widehat{\Delta u}_{t+h} = f(X_t)$$

De voorspelde waarde voor het werkloosheidspercentage in maand $t+h$ is dan:

$$\hat{u}_{t+h} = u_t + \widehat{\Delta u}_{t+h}$$

Specificatie van de BVAR:

$$u_{t+h} = f(X_t) + \varepsilon_t$$

Waarbij:
$u_{t+h} = het\ werkloosheidspercentage\ aan\ het\ eind\ van\ de\ voorspelhorizon\ (t+h)$
$f(X_t) = functie\ geschat\ met\ Bayesian\ VAR\ (BVAR)$
$\varepsilon_t = voorspelfout\ in\ mutatie\ van\ het\ werkloosheidspercentage$

X_t bevat van alle voorspellers zes vertraagde termen in niveaus:

$$X_t = (V_{t-1}, V_{t-1}, V_{t-2}, V_{t-3}, V_{t-4}, V_{t-5}, V_{t-6})$$

$V = vector\ van\ de\ 9\ voorspellers$

De voorspelde waarde voor het werkloosheidspercentage in maand $t+h$ is:

$$\hat{u}_{t+h} = f(x_t)$$

Berekening gemiddelde absolute voorspelfout:

Het verschil tussen de voorspelde en gerealiseerde waarde is de voorspelfout bij een voorspelhorizon van h, in maand t:

$$\rho_{h,t} = \hat{u}_{t+h} - u_{t+h}$$

De gemiddelde absolute voorspelfout is:

$$MAD_h = \frac{\sum_{t=jan\ 2008}^{t=mei\ 2019} |\rho_{h,t}|}{T}$$

Met:

$T = aantal\ maanden\ tussen\ januari\ 2008\ en\ mei\ 2019 = 137$