# Shortlist

Een shortlist met mogelijk relevante afhankelijke variabelen met daarbij per variabele twee kenmerken…

a. De relevantie van de afhankelijke variabele: een nadere uitleg over welke beslissing(en) je kan nemen als je nu al weet welke waarde(n) deze komende maand/kwartaal/jaar gaat hebben.

b. Een eerste inschatting van bijpassende onafhankelijke variabelen: “op basis van welke onafhankelijke variabelen kan naar jouw verwachting de afhankelijke variabele het beste voorspeld worden?”.

## NIET

De volgende tabellen zal niet op worden getraind:

* ad\_JobCandidate
* ad\_EmployeeDepartementHistory
* ad\_Shift
* ad\_SalesPerson (Sales)
* northwind\_CustomerCustomerDemo
* northwind\_CustomerDemographics
* ad\_SalesTerritoryHistory (Sales)
* ad\_SalesPersonQuotaHistory (Sales)
* ad\_TransactionHistoryArchive (Production)
* ad\_ProductReview (Production)
* product
* territory
* supplier
* department
* currency
* store
* BusinessEntityAddress
* Document
* PURCHASE\_ORDER\_DETAIL
* person
* BusinessEntity
* PRODUCTPHOTO

## 

## Voorspellen

### EmployeePayHistory Rate

* Reserveren van geld voor toekomstige salaris verhogingen
* RateChangeDate, om te zien wanneer het salaris veranderd is.
* Combineren met Employee om beter inzicht te krijgen in een typisch salaris. Hiervoor kunnen Employee.BirthDate, Employee.OrganizationLevel, Employee.MaritalStatus en Employee.SickLeaveHours relevant zijn.
* aenc\_employee bevat niet dezelfde informatie. We zullen los moeten encoderen of iets als MaritalStatus een nullwaarde is of niet.

### ProductListPriceHistory Listprice

* Inspelen op prijs stijgingen
* Startdate, enddate, productid. Start en End date voor de wanneer. Productid koppelt de prijs aan een bepaald product/ Verkoopcijfers gelinkt aan de medewerker, om inzicht te krijgen in waarom de medewerker een bonus krijgt.

### ProductCostHistory StandardCost

* Reserveren van geld voor stijgende kosten, productprijzen op tijd aanpassen bij stijgingen en dalingen
* ProductID om te koppelen aan het product.

### TransactionHistory ActualCost

* Voorspellen hoeveel toekomstige transacties gaan kosten
* Poductid, Quantity, TransactionType

### TransactionHistory Quantity

* Voorspellen hoeveel producten verkocht gaan worden
* Poductid, ActualCost, TransactionType

### SalesTerritoryPerson Sales\_ytd

* Voorspellen van verkopen per regio, belangrijk om als factor te gebruiken bij sales.
* Country\_region\_code, sales\_quota

### SalesTerritoryPerson Cost\_ytd

* Voorspellen van kosten per regio
* Country\_region\_code, sales\_quota

### Employee\_bonus\_amount

* Weten hoeveel geld er gereserveerd moet worden om de bonussen uit te kunnen betalen.
* Geen idee eigenlijk
* Deze is makkelijk. Eigenlijk moeten alle bonussen naar de baas gaan. De rest van de werknemers moeten maar minder avocado toast eten ;)
* Nee, maar even serieus. We kunnen kijken of bepaalde employees recent meer bonussen hebben gekregen of zullen krijgen. Deze werknemers zijn dus of hard aan het werken of een alarmbel voor mismanagement van geld.

### SpecialOffer Discount\_pct

* Voorspellen hoe hoog de korting moet zijn. Beter te combineren met andere tabellen om te kijken of kortingen het echt waard zijn en ervoor zorgen of de verkopen hoger zijn.
* offer\_type, min\_quantity, max\_quantity

### ORDERHEADER Purchasing order\_quantity

* Voorspellen hoeveel van een product ingekocht moet worden
* unit\_price, product\_id, stocked\_quantity

### OrderHeader\_Sales order quantity

* voorspellen hoeveel producten verkocht gaan worden, handig voor inkopen
* product\_id, unit\_price\_discount, territory\_id

### WorkOrder ActualCost

* Projecten kosten vaak meer dan van tevoren gedacht. Door het voorspellen van de daadwerkelijke kosten kan van tevoren een ander, hopelijk realistischer, hoeveelheid geld gereserveerd worden.
* plannedcost

### Billofmaterial per\_assembly\_quantity

* Voorspellen van de productie capaciteit, hierdoor productie beter kunnen uitspreiden.
* bom\_level, component\_id, product\_assembly\_id

## ProductVendor OnOrderQTY

* Mergen met product
* Voorspellen van hoeveel producten gekocht gaan worden
* TODO

## Cluster

### Customer

* Overeenkomsten in de klanten vinden
* De address gerelateerde tabellen

### ProductVendor

* Inkoopkosten mergen met Product
* Het categoriseren van ProductVendors
* Waarschijnlijk verkopers van producten aan United Outdoors, niet United Outdoors aan klanten