Use Case – Regression mit RapidMiner

In der Wirtschaft und Industrie gibt es oftmals das Problem, dass numerische Werte vorhergesagt werden müssen wie beispielsweise der Forecast des nächsten Monats/Quartals/Jahres etc. Diese Probleme können mithilfe eines Regressionsmodells gelöst werden. In RapidMiner können wir dies ganz einfach umsetzen. Wie genau zeigt Ihnen der folgende Use Case.

Laden des Datasets

1. Ziehen Sie das „sales_all“-Dataset in den Prozessbereich. Falls Sie dies noch nicht in Ihr Repository importiert haben, folgen Sie bitte der Anleitung aus vorherigen Use Cases.
2. Verbinden Sie das Dataset mit dem „res“-Knoten und schauen Sie es sich genauer an.

Welche Schritte müssen wohl vor einer Regression noch getroffen werden und welches Attribut wird Ihre Zielvariable?

Welches Feature könnten Sie noch hinzufügen?

Daten vorbereiten

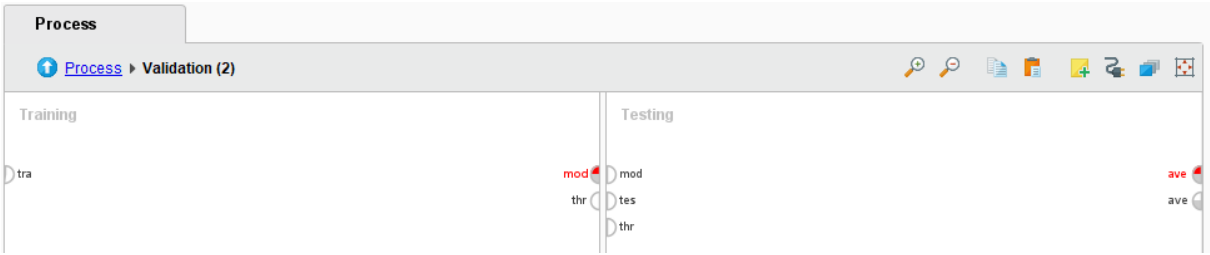
Wie Sie sicher schon vermutet haben muss vor dem Training des Regressionsmodells ein One-Hot-Encoding stattfinden. Da der One-Hot-Encoding Operator in RapidMiner nicht richtig funktioniert, benutzen wir den Nominal to Numerical Operator. Es wird trotzdem ein One-Hot-Encoding durchgeführt.

1. Ziehen Sie den benötigten Operator („Nominal to Numerical) in den Prozessbereich.
2. Verbinden Sie das Dataset mit dem Operator.
3. Wählen Sie mithilfe der Parameter „attribute filter type“ und „attributes“ die umzuformenden Attribute aus und schauen Sie dass „coding type“ auf „dummy coding“ steht.
4. Ziehen Sie den Operator „Set Role“ in den Prozessbereich
5. Verbinden Sie den „Set Role“-Operator mit dem „Nominal to Numerical“-Operator.
6. Wählen Sie als „target role“ → label und wählen Sie bei „attribute name“ Ihre Zielvariable aus.

Regression mit zusätzlicher Split-Validierung

Obwohl das Thema „Evaluierung“ kein Fokus dieser Schulung ist, wird hier bereits ein kleiner Einblick auf die Möglichkeit der Beurteilung des trainierten Modells gegeben.

1. Suchen Sie bei den Operatoren den „Split Validation“-Operator und ziehen diesen in den Prozessbereich.
2. Verbinden Sie den „exa“-Knoten von „Set Role“ mit dem „tra“-Knoten des „Split Validation“-Operators.
3. Klicken Sie nun mit einem Doppelklick auf den Prozess. Sie sollten folgenden Prozess sehen.

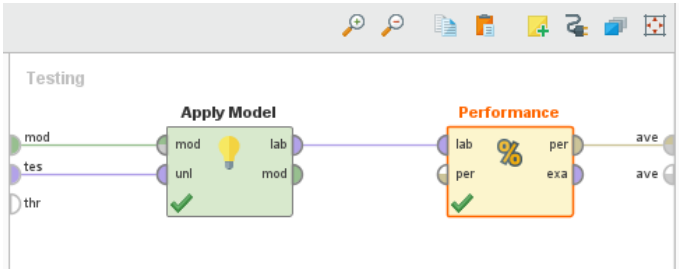


Es handelt sich hierbei einmal um einen Prozess, welcher in einen Trainings- und Testteil aufgeteilt ist. Für den Testteil wird ein gewisser Anteil der Daten vom Example-Set getrennt, welcher nicht für das Training verwendet wird.

Wofür denken Sie kann diese Aufteilung sinnvoll sein?

4. Ziehen Sie nun den „Linear Regression“-Operator in den Trainingsteil des Prozesses und verbinden Sie die „tra“ und „mod“-Knoten miteinander.

5. Wählen Sie den „Apply Model“-Operator aus und ziehen diesen in den Testteil des Prozesses. Verbinden Sie hier das zuvor trainierte Model „mod“ mit dem „mod“-Knoten des „Apply Model“-Operators. Nun müssen Sie noch die Daten auswählen, welche vorhergesagt werden sollen, in diesem Fall die Testdaten. Verbinden Sie hierfür den „tes“-Knoten und den „unl“-Knoten.
6. Ziehen Sie nun den „Performance(Regression)“-Operator in den Prozessbereich und verbinden den „lab“-Knoten mit dem „lab“-Knoten des Operators. Achten Sie darauf, dass bei den Parametern als „main criterion“ → root_mean_squared_error ausgewählt ist und der Haken bei „root mean squared error“ ausgewählt ist.
7. Verbinden Sie zum Abschluss den „per“-Operator mit dem „ave“-Operator.



8. Verlassen Sie den Prozess über den blauen Pfeil neben dem Prozess Symbol in der oberen linken Ecke des Prozessbereichs.
9. Passen Sie nun die „split ratio“ des „Validation“-Operators auf 0.8 an und geben Sie die Knoten „mod“ und „ave“ aus.

Auswertung

root_mean_squared_error

```
root_mean_squared_error: 3.285 +/- 0.000
```

Ihr Modell hat wahrscheinlich einen ähnlichen Wert. Als Wiederholung: Der root_mean_squared_error berechnet sich aus der durchschnittlichen quadrierten Abweichung aller vorhergesagten Datenpunkten zu den tatsächlichen Datenpunkten. Durch die Wurzel (root) wird die Quadrierung aufgehoben.

Was bedeutet dieser Wert also für Ihre Vorhersage?

Warum müssen die Werte beim Root-Mean-Squared-Error quadriert werden?

Warum kann das Regressionsmodell nicht geplottet werden?

Zusatz: Regression mit Decision Trees

Auch mit Decision Trees kann eine Regression erfolgen, obwohl dies eigentlich eine Methode für die Klassifikation ist. Zum Test ersetzen Sie einfach den „Linear Regression“-Operator mit dem „Decision Tree“-Operator. Bevor Sie den Decision Tree trainieren können wählen Sie für den Parameter „criterion“ → „least square“ und „max depth“ → 10.

Wenn Sie nun den Prozess ausführen, sollten Sie einen trainierten Decision Tree ausgegeben bekommen.

Achtung: Decision Trees neigen sehr stark zu **Overfitting** und sollten nicht ohne Train-/Test-Split genutzt werden. Was Overfitting genau ist werden Sie in den nächsten Vorlesungen erfahren.