Projektarbeit: Lineare Regression

Welches Skalenniveau liegt bei welchem Feature vor?

- **Hersteller**: Nominalskala
- **Kilometer**: Verhältnisskala
- **Zylinder**: Verhältnisskala
- **Liter**: Verhältnisskala
- **Tueren**: Verhältnisskala
- **Verhandlungsbasis**: Nominalskala
- **Privatverkauf**: Nominalskala
- **Finanzierung**: Nominalskala
- **Preis**: Verhältnisskala

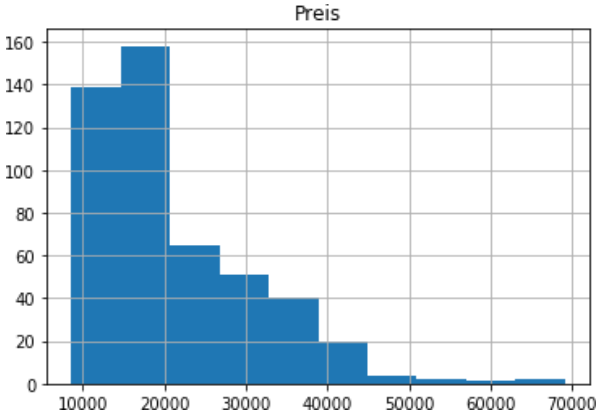
Gibt es fehlende Werte in dem Datensatz?

- **Testdatensatz**: Verhandlungsbasis → 49
- **Trainingsdatensatz**: Verhandlungsbasis → 55

Beschreiben Sie die Beziehung der Features untereinander.

	Kilometer	Zylinder	Liter	Tueren	Verhandlungs-basis	Preis
Kilometer	1.000000	-0.022081	-0.013209	-0.042549	0.129348	-0.123950
Zylinder	-0.022081	1.000000	0.957167	0.013424	-0.340372	0.582690
Liter	-0.013209	0.957167	1.000000	-0.080300	-0.380444	0.570228
Tueren	-0.042549	0.013424	-0.080300	1.000000	0.129354	-0.129523
Verhandlungs-basis	0.129348	-0.340372	-0.380444	0.129354	1.000000	-0.258011
Preis	-0.123950	0.582690	0.570228	-0.129523	-0.258011	1.000000

Beschreiben Sie die Verteilung der Zielgröße. Ist diese symmetrisch?



- Nicht symmetrisch sondern rechtsschief

Beschreiben Sie die Beziehung der Features zu der Zielgröße

- **Korrelation Kilometer und Preis**: -0.12395018175966957
- **Korrelation Zylinder und Preis:** 0.5826898919493045
- **Korrelation Liter und Preis:** 0.5702277959303835
- **Korrelation Tueren und Preis:** -0.12952291670709937
- **Korrelation Verhandlungsbasis und Preis:** -0.2580113780550638

Wie viele Ausprägungen umfassen die nominalen Features?

- **Privatverkauf*:** 2 Ausprägungen
  - Ja: 163
  - Nein: 319
- **Finanzierung*:** 2 Ausprägungen
  - Ja: 183
  - Nein: 299
- **Hersteller*:** 6 Ausprägungen
  - BMW: 50
  - VW: 196
  - Renault: 44
  - Ford: 66
  - Fiat: 36
  - Daimler: 90
- *Bezieht sich auf den Trainingsdatensatz

Handelt es sich um ein nominales Feature? Wie wurde dieses Feature transformiert?

- **Verhandlungsbasis*:** 2 Ausprägungen (NaN-Werte vorhanden)
  - 0.0 (Nein): 210
  - 1.0 (Ja) 217
- Transformation (binär evtl. via One-Hot-Encoding)
  - Ja → 1.0
  - Nein → 0.0
- *Bezieht sich auf den Trainingsdatensatz

Informieren Sie sich, was man unter einem One-Hot-Encoding versteht.

- **Definition:** One-Hot-Encoding ist ein Prozess, bei dem kategorische Variablen in eine Form umgewandelt werden, die ML-Algorithmen zur Verfügung gestellt werden können, um eine bessere Güte bei der Vorhersage zu leisten.
- **Genauer:**
  - Mit One-Hot-Encoding werden nicht metrische Features in eine Form überführt, die von Machine Learning Algorithmen verarbeitet und interpretiert werden können.
  - Analog zur Einführung von Dummy Variablen in der Statistik, wird jede Ausprägung eines kategorialen Features in eine separate Spalte mit binärem Wertebereich überführt (1/0 bzw. True/False). Ein Feature mit n unterschiedlichen Ausprägungen wird nach dem One-Hot-Encoding durch n neue Spalten repräsentiert, von denen pro Datensatz genau eine Spalte den Wert 1 enthält und alle anderen (n-1) Spalten eine 0 aufweisen.
  - → Daher der Name One Hot
- Bpsw. "Verhandlungsbasis" mit Ausprägungen "Ja" und "Nein" in 1.0 bzw. 0.0. (So auch bei "Hersteller", etc. sinnvoll)

Das Feature Verhandlungsbasis weist fehlende Werte auf.

Entwickeln Sie zunächst ein Verständnis für diese Denkweise: Überlegen und begründen Sie, welchen Wert Sie den fehlenden Werten zuweisen möchten

- **1. Möglichkeit**
  - Man könnte überlegen, die fehlenden Werte mit dem Wert zu überschreiben, der am häufigsten vorkommt
  - → Das "Problem" bei diesem Datensatz, ist jedoch, dass die Werte nahezu gleichverteilt sind und somit ein "häufigster" Wert nicht existiert. Diese Herangehensweise scheidet somit aus.
- **2. Möglichkeit**
  - Die fehlenden Werte werden mit dem Wert überschrieben, der betriebswirtschaftlich mehr Sinn macht.
  - Verhandlungsbasis: Ja → 1

    - Verhandlungsbasis: nein --> 0
- **Überlegung:**
    - Ein Verkäufer ist eher genervt, wenn er nicht verhandlungsbereit ist und VB = 1 gesetzt ist, als dass es ihn stört, wenn die Preisvorstellung akzeptiert wird. Für eventuelle Nachverhandlungen steht er in der Regel trotzdem zur Verfügung.
    - → Daher: Überschreiben der Werte mit 0

Alternativ können Sie in Betracht ziehen, ein Feature mit fehlenden Werten zu löschen, also von ihren Analysen auszuschließen. Welche Nachteile könnte das haben?

- **Fakten:** Korrelation Verhandlungsbasis und Preis: -0.2580113780550638 (schwache negative Korrelation)
- Beim Ausschluss von Werten, welche hoch mit der Zielgröße korrelieren, könnte die Präzision der Prognose leiden.
- **Hier:** Geringe Korrelation lässt Ausschluss grundsätzlich sinnvoll erscheinen

Ebenso können Sie Instanzen löschen, die fehlende Werte aufweisen. Ist das ratsam?

- Generell bei überschaubarer Datensatzgröße nicht zu empfehlen
- In diesem Fall würden 55 von insgesamt 482 Instanzen gelöscht werden und die zugrundeliegende Datenbasis des Datensatzes um über 11% verringert werden.
- Neben den fehlenden Werten gehen zudem auch alle übrigen korrekt und unter Umständen mühsam erfassten Werte verloren.

Prognose

Einwirkungen der obigen Erkenntnisse auf den Sachverhalt

1. **Skalenniveaus**
    a. Relativ unproblematisch, da viele Merkmale Verhältnisskaliert sind
    b. Nominale Merkmale müssen jedoch interpretierbar gemacht werden
2. **Fehlende Einträge**
    a. Fehlende Einträge stellen Problem bei Durchführung der Regression dar, weshalb wir vor dem Hintergrund unserer Annahme bezüglich des Zwecks dieser Analyse auf die oben getroffene Entscheidung zurückgreifen
    b. Fehlende Werte (Verhandlungsbasis) werden mit „Nein“ bzw. 0.0 gefüllt
3. **Verteilung der Zielgröße**
    a. Die Verteilung der Zielgröße im Trainingsdatensatz gibt uns insofern einen Anhaltspunkt, dass die Verteilung der Zielgröße im Testdatensatz ähnlich aussehen sollte
    b. Dabei unterstellen wir, dass die Einträge des Trainings- und Testdatensatzes zufällig aus einem gemeinsamen Datensatz gezogen wurden und sich die Verteilungen somit grundsätzlich ähnlich sind
4. **Beziehung der Features untereinander und zur Zielgröße**
    a. Anhand der Beziehungen der Features untereinander bzw. viel mehr anhand der Beziehung der Features zur Zielgröße lässt sich eine „manuelle“ Feature-Selektion vollziehen
    b. Features, die kaum merklich mit der Zielgröße korrelieren können vor dem Hintergrund der Modellkomplexität vernachlässig werden
    c. Für die Identifizierung relevanter bzw. nicht relevanter Features verwenden wir eine Heatmap in welcher die Beziehungen aller Features untereinander visualisiert sind
    d. Die von uns getroffene Auswahl der Features ist tendenziell subjektiv geprägt, da wir selbst Features einbringen, welche eine sehr schwache Korrelation mit der Zielgröße aufweisen; sehr schwach korrelierte Features mit einem Korrelationskoeffizienten nahe Null eliminieren wir jedoch aus unserem Datensatz
    e. Die „manuelle“ Auswahl umfasst somit Kilometer, Zylinder, Liter, Tueren, Verhandlungsbasis, BMW, Daimler, Fiat, Ford, Volkswagen, Preis (*Entfernt wurden Privatverkauf, Finanzierung und Renault, da diese quasi gar nicht mit der Zielgröße korrelieren*)

5. **Ausprägungen der Merkmale und One-Hot-Endcoding**
    a. Einige der nominalen Merkmale sind im Gegensatz zum Merkmal „Verhandlungsbasis“ noch nicht kodiert, weshalb wir eine Kodierung vornehmen müssen
    b. Dafür verwenden wir das sogenannte One-Hot-Encoding, welches uns die Merkmalsausprägungen binär transformiert
    c. Somit lassen sich alle Merkmale des Datensatzes für unsere Prognose nutzen

Vorgehen

1. **Definitionen von Hilfsfunktionen für die lineare Regression**
    a. Definition One-Hot-Encoding Funktion, welche das One-Hot-Encoding durchführt
    b. Definition Funktion für ein Regressionsmodell, welche die Regression mit n erklärenden Variablen durchführt
2. **Durchführung von (multivariaten) linearen Regressionen**
    a. Verwendung von einer bis 8 erklärenden Variablen (Polynom 8er Ordnung) und Dokumentation des jeweiligen MSE
    b. Auswahl des besten Modells vor dem Hintergrund eines zu minimierenden MSE
    c. Run und Submit der Ergebnisse des besten Modells (3 erklärende Variablen)
        ➔ Kaggle Score: 0.92166
3. **Weitere Überlegungen**
    a. Da im Punkt 2 alle vorhandenen Features verwendet wurden, haben wir uns überlegt eine „manuelle“ Feature-Selektion vorzunehmen (siehe Beziehung der Features untereinander und zur Zielgröße)
        ➔ Verwendung einer Heatmap
    b. Durchführung von (multivariaten) linearen Regressionen mit manuell ausgewählten Features
    c. Run und Submit der Ergebnisse des besten Modells (2 erklärende Variablen)
        ➔ Kaggle Score: 0.97180
4. **Anwendung lineare Lasso-Regression**
    a. Definitionen von Hilfsfunktionen
        i. Funktion für das Einlesen, bereinigen und One-Hot-Encoden der Daten, die die Daten einliest, bereinigt und One-Hot encoded
        ii. Funktion zur Erstellung des Modells, die eine Gittersuche durchführt und den besten Hyperparameter alpha zurückgibt
    b. Durchführung von (multivariaten) linearen Lasso-Regressionen
    c. Run und Submit der Ergebnisse des besten Modells (2 erklärende Variablen)
        ➔ Kaggle Score: 0.97154
5. **Erreichung des Top-Scores unserer Submissions**
    a. Durch splitten des originalen Datensatzes in einen Trainings- und einen Testdatensatz erreichen wir zufällig noch einen marginal besseren Kaggle-Score.
    b. Dies erklären wir uns damit, dass zufällig bei diesem Random-State die Verteilung des Trainingsdatensatzes ähnlicher zu derer des in Kaggle derzeit hinterlegten Validierungsdatensatzes.
    → Der Score beträgt bei Run dieses Modells 0,97310.

Lessons-Learned

- Manuelle Feature Selektion erbrachte sowohl hinsichtlich der Rechenzeit auch im Hinblick auf die Prognosegüte Vorteile
- Anwendung der linearen Lasso-Regression wirkte sich nicht mehr positiv auf die Prognosegüte aus, selbst wenn nur die manuell selektierten Features eingebracht wurden
- Insgesamt sehr zufriedenstellendes Ergebnis