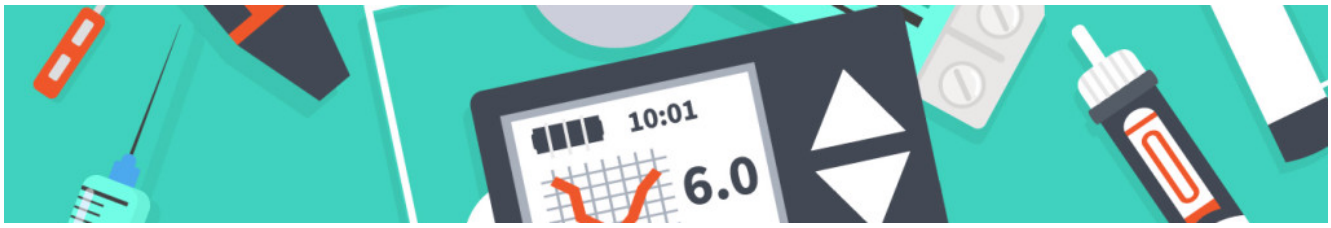
Data Wrangling (daw)
Cédric Huwyler

Fachhochschule
Nordwestschweiz

Minichallenge: Diabetes bei den Pima-Indianerinnen



1 Ausgangslage

Du bewirbst dich als Data Scientist bei einem Startup im Gesundheitsbereich, das eine App zur Risikoabschätzung häufiger Alterskrankheiten entwickeln möchte. Die App soll die User einerseits regelmässig über die abgeschätzten Risiken in Verbindung mit dem eigenen Lebensstil informieren und ab gewissen Schwellwerten auch Warnungen herausgeben und frühzeitig Übungsprogramme und Arztbesuche empfehlen.

Du schaffst es in die zweite Runde und erhältst ein Datenset, das du säubern, verstehen, für erste Erkenntnisse verwenden und geeignet auf eine Modellierung vorbereiten sollst. Am Gespräch sollst du dann deine Erkenntnisse kurz präsentieren und deine Ansätze motivieren. Das Datenset handelt über das Vorkommen von Diabetes bei Frauen des Pima-Indianerstamms in der Nähe von Arizona, USA. Der Stamm ist für die Diabetesforschung unter anderem interessant, weil mit der schnellen Änderung der Lebensumstände Diabetes in den letzten Jahrzehnten stark aufgekommen ist.

Du freust dich, dass du deine Skills zeigen kannst und machst dich sofort an die Arbeit. Natürlich wäre es in einem Bewerbungsgespräch die Idee, dass du die Aufgabenstellung selbstständig erarbeitest, hier wirst du aber ein wenig geführt, so dass du möglichst vielen Lernergebnissen kommst.

2 Aufgabenstellung

1. **Daten einlesen**

   a) Lies die Daten aus dem beigelegten Excel-File 'diabetes.xlsx' ein. Lies dazu auch die beigefügte Beschreibung der einzelnen Merkmale und versuche diese ganz grob zu verstehen.

   b) Welche Merkmale sind diskret, welche stetig? Handelt es sich konkret um nominal-, ordinal-, intervall- oder verhältnisskalierte Merkmale?

2. **Exploration**

   a) Untersuche zuerst die Verteilungen der einzelnen Merkmale, jedes für sich. Gibt es auffällige Werte, die auf Datenfehler, Ausreisser oder fehlende Werte hinweisen?

   b) Stelle sicher, dass alle fehlenden Werte sauber mit '`NaN`' markiert sind und stelle die Verteilung der einzelnen Merkmale danach noch einmal graphisch mit Histogrammen dar.

   c) Wie hängen die Merkmale mit der Zielvariable '`class`' zusammen? Fallen dir Merkmale auf, die bereits klare Hinweise auf das Diabetesrisiko geben? Benutze eine geeignete graphische Darstellung.

   d) Gibt es Merkmale, die untereinander stark korrelieren? Visualisiere dazu die Daten mit einer *Scatter Matrix* und berechne die *Korrelationsmatrix*.

      Hinweis: Die Zielvariable '`class`' muss bei dieser Analyse nicht mehr vorkommen.

3. **Fehlende Werte**

   a) Quantifiziere die Anzahl der fehlenden Werte pro Merkmal absolut und in Prozent.

   b) Hängen die fehlenden Werte mit der Zielvariable '`class`' zusammen? D.h. würde bereits das Fehlen oder Nicht-Fehlen eines Merkmals auf ein erhöhtes oder vermindertes Diabetesrisiko hinweisen?

   c) Hängt das Fehlen eines Merkmals vom Fehlen eines anderen Merkmals ab?

      Hinweis: Hier kannst du den Zusammenhang zwischen fehlenden Werten zum Beispiel grob prüfen, indem du überall im Data Frame eine Null setzt, wo der Wert nicht fehlt, und eine Eins setzt, wo der Wert fehlt, und dann die Korrelationsmatrix berechnest. Das Mass passt zwar nicht perfekt, aber für erste Analysen reicht es aus.

   d) Zum typischen Data Wrangling gehört die Evaluation von Strategien zum Umgang mit fehlenden Werten. Eine Strategie ist das Entfernen aller Zeilen mit mindestens einem fehlenden Wert. Wieviele Zeilen müsstest du hier konkret aus dem Datensatz entfernen?

   e) Statt sie zu entfernen möchten wir die fehlenden Werte lieber imputieren. Argumentiere hier kurz, ob du ein Ersetzen mit Durchschnitt oder Median besser findest und erstelle dann ein imputiertes Data Frame.

   f) Statt jedes einzelne Merkmal für sich zu imputieren, kann auch eine modellbasierte Imputationsstrategie benutzt werden, die die Merkmale im Zusammenhang zueinander betrachtet. Suche dir eine solche Strategie aus (zum Beispiel `KNNImputer` aus scikit-learn) und erstelle ebenfalls ein imputiertes Data Frame damit.

4. **Erste Erkenntnisse gewinnen**

   Meistens möchtest du nicht gleich als erstes ein Modell trainieren, sondern zuerst einige grundlegende Zusammenhänge in deinem Datenset verstehen, damit du nachher auch die Resultate der Modellierung verstehst. Im Folgenden dazu einige Vorschläge:

   a) Was ist das mittlere Diabetesrisiko auf die Anzahl der bereits erlebten Schwangerschaften?

      Hinweis: Die Zuverlässigkeit deiner Aussage nimmt mit abnehmender Gruppengrösse ab. Beachte diese Eigenschaft, wenn du allgemeine Aussagen machst.

   b) Die Diabetes Pedigree Function (dpf) berechnet sich aus dem Auftreten von Diabetes im Stammbaum. Ist sie alleine ein klarer Indikator für das Auftreten von Diabetes im eigenen Leben? Benutze für deine Untersuchung einen *Density-Plot*.

   c) Wie wirkt sich das Alter auf das Diabetesrisiko aus? Führe hier Altersklassen ab 20 im Abstand von 10 Jahren ein und berechne das durchschnittliche Diabetes-Risiko.

      Hinweis: Die Zuverlässigkeit deiner Aussage nimmt wie in a) mit abnehmender Gruppengrösse ab. Beachte diese Eigenschaft, wenn du allgemeine Aussagen machst.

   d) Ebenfalls möchten wir gerne wissen, wie stark Fettleibigkeit zum Diabetesrisiko beiträgt. Erstelle dazu eine neue Spalte '`bmi_class`' mit der folgenden Einteilung:

$$\text{bmi\_class} = \begin{cases} \text{'underweight'}, & \text{bmi} < 18.5 \\ \text{'normal'}, & 18.5 \leq \text{bmi} < 25 \\ \text{'overweight'}, & 25 \leq \text{bmi} < 30 \\ \text{'obese'}, & \text{bmi} \geq 30 \end{cases}$$

      Die Spalte soll über die Ordinalskala der BMI-Klasse informiert sein. Berechne nun das mittlere Diabetesrisiko und die Anzahl der Probanden pro BMI-Klasse. Was ist deine Schlussfolgerung?

   e) Was ist der Einfluss des Bluthochdrucks auf das Diabetesrisiko? Argumentiere auch hier zum Beispiel mit einem Density-Plot.

5. **Eine kleine Datenpipeline**

   Da du nun relativ viel über den Datensatz weisst, möchtest du dich gerne ans Modellieren machen. Du hast dir als Benchmarks die Modelle *Lineare Regression* und *Random Forest* vorgenommen. Beide Modelle können in Scikit-Learn nicht oder nur schlecht mit fehlenden Werten umgehen, darum möchtest du prüfen, welche Imputationsstrategie für diesen Fall die beste ist.

   a) Schreibe eine Funktion `preprocess( xlsx_path, missing_value_strategy )`, die dir das Excel-File aus `xlsx_path` einliest, die fehlenden Werte entsprechend markiert, die Daten zufällig in ein Trainings- und Validierungsset einteilt und dann gemäss `missing_value_strategy` behandelt und am Schluss ein Data Frame zurückgibt. Im Prinzip geht es hier darum, deine Schritte aus den Teilen 1 und 3 in eine Data Preprocessing Pipeline zu integrieren. Evaluiere die folgenden Strategien: fehlende Werte entfernen ('drop'), fehlende Werte mit Mean imputieren ('mean'), fehlende Werte mit Median imputieren ('median') und fehlende Werte modellbasiert imputieren ('model'). Kommentiere deinen Code ausreichend und beachte, dass das Validierungsset natürlich nur mit Informationen aus dem Trainingsset imputiert werden darf.

   b) **(für Fortgeschrittene)** Spielt die gewählte Strategie zur Fehlerbehandlung für die beiden gewählten Modelle eine Rolle? Prüfe grob, indem du die Modellperformances für die verschiedenen Strategien evaluierst. Für eine korrekte Überprüfung bräuchtest du hier eine Pipeline mit Cross Validation, darauf soll aber dann in der Kompetenz *Supervised Learning* eingegangen werden.