Final Presentation

What should be said.

Stanislav Arnaudov

November 7, 2018

Hallo, mein Name ist Stanislav und heute präsentiere ich die Ergebnisse meiner Bachelorarbeit.

1 Motivation

Zunächst, kurz zu Motivation. Es gibt viele Anwendungen, wo man irgendwas messen will. Hier kommen die Sensor Messnetzwerke ins Spiel. In unserem Fall ist das Netzwerke heterogen, spricht es enthält Sensoren von unterschiedlichen Qualitäten. Mit "unterschiedlichen Qualitäten" ist gemeint, dass die "guten" Sensoren präzise und zuverlässige Daten liefern (und zwar wir wissen das). Über die "schlechten" Sensoren können wir dagegen keine Aussagen machen. Also, wir wissen nicht was für Werte sie liefern.

Damit kommen einige interessante Fragen:

- kann man die schlechten Sensoren verwenden, um die Werte von den guten zu vorhersagen?
- kann man die besonders schlechten Sensoren im Netz identifizieren?

Und wie haben wir diese Ziele versucht zu erreichen? Bei Untersuchung von einigen stochastische Regressionsmodelle und unter Verwendung von der so genannten Feature Importance Technik.

2 Daten

Um das Problem zu konkretisieren, sagen wir auch etwas zu den konkreten Daten. Es geht um ein Messnetzwerk von Feinstaubsensoren in Stuttgart. Da gibt's zwei Typen von Sensoren.

- 3 Sensoren von hoher Qualität. Die Daten davon sind uns von LUBW zur Verfügung gestellt.
- außerdem gibt's auch ein Netz von zahlreichen billigen DIY Sensoren, dessen Daten öffentlich auf luftdaten.info zu finden sind.

Bezüglich Zeitraum - wir haben das Jahr 2018 betrachtet.

Alle Sensoren liefern zwei Arte von Werten - PM10 und PM2.5. Diese sind verschiedene Merkmale, mit denen Feinstaub beschrieben wird.

Es gab ganz viele Probleme mit den Daten. Z. B. die luftdaten Sensoren messen jede Minute, aber die LUBW liefern über 30 Minute gemittelte Werte. Deswegen müssten wir das synchronisieren. Wir haben uns entscheiden drei verschiedene Mittelungintervale zu wählen, damit wir sehen kann, wie die Ergebnisse entsprechend variieren. Wir haben Tagesmittel, 12 Stunden Mittel und eine Stunde Mittel untersucht. Das heißt, von den Daten haben wir im Endeffekt 3 Sets generiert.

Bemerkung zu Station SAKP - die Daten davon waren nicht vollständig. Für die erste 3/4 des Jahres sollten wir die Feinstaubwerte interpolierten. Wir haben den Sensor trotzdem betrachten, damit wir sehen kann, was für schlechte und falsche Ergebnisse zu beobachten sind.

3 System

Grob gesehen, was wollten wir bauen?

Unter Verwendung von Daten von allen andren Sensoren, wir wollen die Feinstaubwerte von einem von den LUBW Sensoren vorhersagen. Wir betrachten immer den Fall, wo man Daten aus der Vergangenheit verwendet, um Werte in der Zukunft zu vorhersagen. Wie gesagt, die LUBW Sensoren sind die zuverlässigen, deswegen betrachten wir die als Groundtruth.

4 Ziele und nicht Ziele

Bei unseren Zielen, gehen wir davon aus, dass wir keine große Menge von Daten haben. Daher, unser Ziel war nicht das beste Model für den Fall zu bauen, sondern eher zu zeigen, dass die untersuchten Modelle besser als ein Baseline sind. Unsere Arbeit ist in diesem Sinn explorativ. Wir haben nicht die Hyperparameter von den Modellen getweaked um die besten Ergebnissen zu erhalten, sondern die Modelle fest ausgewählt und gesagt "Ok, das untersuchen wir, was für Ergebnisse können diese erreichen".

Das andere Ziel war, die relevanten Features für die Modelle herausfinden. Hier gibt's ein Sternchen. Warum? Die Wichtigkeit von einem Feature ist nur im Kontext von dem entsprechenden Modell. Man kann nicht wirklich sagen "dieser Sensor ist in der Realität unwichtig". Der ist eventuell unwichtig nur für das Modell, das ihn verwendet. Anderseits, wenn ein Sensor konsistent von einigen Modellen als unwichtig gesehen ist, dann können wir zumindest vermuten "Ok, mit dem Sensor ist etwas los."

5 Stochastische Regressionsmodelle

Was genau sind stochastische Regressionsmodelle? Regressionsmodelle (also man versucht einen reellen Wert zu vorhersagen), die eine Wahrscheinlichkeitsverteilung für jeden Eingabepunkt generieren. Also, bei diesen Modellen die tatsächliche Beobachtung wird durch Verteilung modelliert. Zu bemerken ist, dass die Verteilung für jeden Eingabepunk generiert ist. Das kann man hier sehen. Das ist ein Beispiel für solch ein Modell.

Die blaue Kurve ist die Sinus Funktion und die großen blauen Punkten sind die Beobachtung davon, also mit diesen wird das Modell trainiert. Danach wird das Modell an jeder Stelle evaluiert. Die dicke rote Kurve ist der Erwartungswert von der Verteilung. Außerdem werden aber überall gewisse Wahrscheinlichkeiten gegen, dass die Beobachtung da ist. Man kann sehen, die Verteilungen sind breiter in den Bereichen weit weg von den Beobachtungen. Da ist das Modell unsicher was der Wert von der Funktion ist.

6 Konkrete Modelle

Welche konkrete stochastische Regressionsmodelle haben wir Untersucht.

6.1 BNN

Zunächst, Bayesian Neuronale Netze. Sehr ähnlich zu gewöhnlichen neuronalen Netzen, aber die Gewichten sind Wahrscheinlichkeitsverteilungen. In unserem Fall - Normalverteilung. Damit man das Netz mit einem Eingabepunkt evaluieren kann, muss man von diesen Verteilung zunächst je eine Realisierung ziehen und danach die Forward-Propagation durchführen. Wenn man dies N-mal mit demselben Eingabepunkt macht, hat man N-Werte von den generierten Verteilung gezogen. Also, bei BNNs erhält man die Verteilung nicht explizit, sonder in der Form von vielen gezogenen Realisierung.

6.2 MDN

Mixture Density Networks. Man hat hier eine Mixtur von gewichtete Verteilungen. In unserem Fall sind diese Normalverteilungen. Die Gewichte von der Mixtur sind Mixing coefficiants genannt. Jede einzelne Verteilung hat eigene Parameter - in unseren Fall Erwartungswert und Varianz. Die Parameters und die Mixing coefficiants sind Funktionen von der Eingabe und diese Funktionen werden durch ein neuronales Netz modelliert. Man trainiert dann das Netz so, dass die Wahrscheinlichkeit von den Beobachtungen maximiert ist.

Noch mal, wie gesagt, wir sind nicht nach der Suche vom geeignetsten Modell. Wir wollen bestimmte für uns interessante Modelle untersuchen, damit man für die Zukunft weißt "Ok auf diesen Daten, das funktioniert, das funktioniert nicht."

6.3 Empirisches Modell

Das ist ein ganz simples Modell, das wir als Baseline verwenden. Also wir hoffen, dass die BNNs and MDNs besser als dieses sind. Bei dem empirischen Modell verwendet man nur die vorherigen Werte von einem Sensor um die zukünftigen Werte desselben Sensors zu vorhersagen - also keine Features von dem ganzen Netz. Die Werte von der Vergangenheit sind als Realisierungen von einer Zufallsvariable betrachtet. Damit kann man eine Verteilung für die Zukunft schließen.

7 Evaluierung von Wahrscheinlichkeitsverteilungen

Jetzt kommen wir zu die Frage, wie vergleicht man überhaupt eine vorhergesagte Verteilung und eine reellwertige Beobachtung. Dafür verwendet man spezielle Metriken. Von den Bildern sehen wir die offensichtlichen Unterscheiden zu eine Punktschätzung. Der blaue Strich ist die tatsächliche Beobachtung, die wir vorhersagen wollen. Unten ist eine reine Punktschätzung, wo man sich wirklich nur die Distanz dazwischen schauen kann. Wenn die Vorhersage eine Verteilung ist, ist das nicht der Fall. Hier kann man ganz expressivere Aussagen machen.

8 Proper Scoring Rules

Die spezielle Evaluierungsmetriken sind bei uns die "Proper Scoring Rules". Die messen den Fehler zwischen einer Verteilung und einer Beobachtung. "Proper" weil die "Proper Eigenschaft" gilt. Nämlich - die echte Verteilung muss den kleinsten Fehlerwert erhalten. Hier schauen wir nur ein Porper scrolling rule - CRPS oder Continuous Rank Probability Score. CRPS generalisiert den mittleren absoluten Fehler zwischen einer Punktschätzung und einem reellen Wert. Das Scoring Rule betrachtet die ganze Verteilung und nicht nur einen bestimmten Punk davon. Das Bild zeigt intuitiv, wie CRPS ausgerechnet wird. Der Wert von CRPS ist der Quadrat der blauen Fläche hier.

9 Verification Rank Histograms

Verification Rank Histograms sind ein Zeug für visuelle Evaluierung von den generieren Verteilungen. Die Intuition - falls die Verteilung "gut" ist, verhält sich die Beobachtung als eine zufällige von der Verteilung gezogene Realisierung. Dafür generiert man viele Samples von der Verteilung, sortiert die und schaut sich, wo liegt der Beobachtung, in welchem Intervall. Man macht das für alle untersuchten Beobachtungen im Test Set und akkumuliert die Ergebnisse in ein Histogramm. Falls das Histogramm uniform verteilt ist, kann man sagen, dass die Beobachtungen nicht so verscheiden als die Samples von der Verteilung. Das ist auch hier illustriert.

- falls das die Art von den meinten Beobachtung ist, ist das Histogramm uniform.

- falls die Verteilungen zu konzentriert um die Beobachtungen sind, dann kann man diese Spitze im Histogramm merken.
- die Spitze ist an der Seite falls die Verteilung nicht so nah an der Beobachtungen sind.

10 Feature importance

Damit evaluiert man wie viel Information bringt ein bestimmtes Feature dem entsprechenden Model. Wie wird das gemacht - Man Variiert die Eingabe Daten (Feature pro Feature) und merkt sich die Änderung im mittleren CRPS-Score. In anderen Wörtern, wir merken wie viel schlechter wird der Fehler wenn ein Feature "kaputt gemacht ist". Falls die Änderung groß ist, dann schließen wir, dass das entsprechende Feature viel Information für das Model erhaltet und das Model legt eine großen Wert auf das Feature. Falls die Änderung klein ist, ist das Feature dagegen von kleineren Bedeutung für das Model.

Wichtig - Dies misst aber die Informationsgehalt des Features nur im Kontext von dem Modell!

Außerdem, das Modell muss einigermaßen "gut" sein, damit man die Feature Importance Ergebnisse trauen kann.

11 Training

Das Trainieren war im allgemeinen ziemlich kompliziert. Es gab viele unabhängigen Kriterien, wie man ein Modell trainieren konnte und wir wollten alle mögliche Kombinationen untersuchen. Die Kriterien sind:

- Vorhergesagte LUBW Station
- Mittelungintervall der Daten - wir haben gesagt, wir haben 3 Sets mit Daten.
- Verwendung von den anderen LUBW Sensoren. Wir vorhersagen einen Sensors, werden aber die anderen zwei als Features benutzt.
- Feinstaubwert - PM10 oder PM2.5
- Modelltyp - Wir haben insgesamt zwei MDNs und ein BNN utersucht. Dazu kommt noch das empirische Model

Trainiert wurde auf dem SDIL Plattform mit 140 Kerne. BNNs waren aber trotzdem schwierig zu trainieren. Durchschnittliche Rate - 3 bis 4 BNNs pro Tag trainiert und evaluiert. Im Vergleich dazu, alle MDNs waren für 3 Tage fertig.

Hier präsentiere ich nur einen kleinen Teil der Ergebnisse.

11.1 Results: curves

Zunächst schauen wir, wie die Kurvenverläufe von den Modellenausgabe aussehen.

Für intuitive Vorstellung, was das empirische Modell Erreichen kann, so sehen die vorhersagen bei ihm aus. Wir sehen hier, die Verteilungen sind breit und fassen fast alle mögliche Werte um. Man kann vielleicht sagen, dass die Egebnisse gültig sind, sind aber gar nicht aussagekräftig. Anhand von diesen Ergebnissen, kann man wirklich nicht die Beobachtungen akkurat raten.

Anderseits ist das hier die Kurve von einer von den MDNs. Der Unterschied ist klar. Dieses Modell versucht tatsächlich den Beobachtungswert zu approximieren. Da wo der Wert durch den Erwartungswert nicht approximiert werden kann, ist die Verteilung breiter. Damit ist die Beobachtung immer noch modelliert, obwohl da die Unsicherheit groß ist. Natürlich ist das Model nicht perfekt. Auf dem Test Set sind die Ergebnisse deutlich besser, was Overfitting impliziert.

Ganz ähnlich sind die Egebnisse bei den BNNs. Hier sehen wir auch so was wie Overfitting. Bei dem Train Set kann man die blaue Kurve fast nicht sehen, weil das Modell sie ziemlich gut modelliert. Das ist aber nicht der Fall beim Test Set. Da kann man ziemlich viele Stellen sehen, wo dass Modell die Beobachtung nicht raten kann. Trotzdem, ist auch offensichtlich, dass die Verteilungen nicht zufällig sind und einigermaßen die echte Beobachtung entsprechen.

11.2 Results: plots

(One hour plots)

Genauer kann man die Güte von den Modellen mit den nächsten Plots sehen. Mit ihnen verglichen wir direkt einige Modelle. Bemerkung: hier geht's um Modelle, die Daten modellieren, die über eine Stunde gemittelt sind. Weitere Bemerkung: kleinere Scores sind besser. Wir betrachten wie gesagt nur CRPS Werte. Zeilen repräsentieren verschiedene LUBW Stationen, die vorhergesagt sind. Die Spalten sind dagegen die zwei Feinstaub werte - PM10 und PM2.5.

Wenn wir die Station SBC genauer anschauen - schnell kann man merken, dass sogar ohne LUBW Daten die MDNs Modelle besser als das Baseline sind. Die MNDs bleiben besser in allen Fällen. Anderseits, wenn man die Vorhersage für PM10 nachschaut, ist das BNN schlechter als das empirische Model wenn es keine Werte aus LUBW Sensoren als Features verwendet wurden. Unsere Vermutung, BNNs sind quasi "empfindlicher" zu "Informationsverlust".

Die Scores für die SAKP Station, wenn man darüber ein bisschen überlegt, versteht man, dass etwas nicht ganz in Ordnung ist. Erinnerung, SAKP war die "schlechte" LUBW Station ohne vollständige Daten. Im Bild merk man, dass nichts sich ändert. Die MDNs sind besser als das Empirische Modell, das must aber irgendwie zufällig sein, weil wissen, dass diese Station nicht wirklich vorhersagbar ist. Die Daten sind interpoliert. Wenn man die Modell-kurve prüft, versteht man, warum diese Ergebnisse aufgetreten sind. Die Verteilung sind brei und fassen alle "Beobachtungen" (interpolierte Werte) um. Wie beim empirischen Modell, das ist eine wahre Modellierung, die aber uns gar nicht sagt.

11.3 Results: Rank Histograms

Als Nächstes haben wir zwei Rank Histogramms, die uns die Probleme mit den Modellen uns deutlich zeigen. Die Histogramms haben die Form von einem "U". Das heißt, die generierten Verteilung sind entweder komplett links oder rechts von der Beobachtung. In beiden Fällen, liegt die Beobachtung nicht in der Verteilung. Das natürlich ist nicht so schön, aber das ist was die Ergebnisse sind.

11.4 Results: Feature Importance

Letztendlich kommen wir zu Feature Importance. Wieder haben wir jetzt keine Zeit für alle Feature Importance Plots. Das hier ist einer, der Modellen für den PM10 Wert von SBC Vergleicht. Oben sind die Modelle, die die anderen zwei LUBW Sensoren verwenden und unten sind diese, die keine LUBW-Daten nutzen. Auf ersten Blick sieht alles verwirrend aus. Eine Sache, die aber offensichtlich ist - hier ist der Sensor 146 konsistent für alle Modelle von keiner Bedeutung. Das ist in der Tat interessant und die Schlussfolgerung (oder eher die Vermutung) wäre "Der Sensor 146 ist besonders schlecht, weil die Modelle kaum Information von dem entnehmen". Etwas anderes, was merkbar ist, hier oben, all Modelle legen einen großen Informationswert auf dem LUBW Sensor SNTR. Das ist natürlich zu erwarten. Was nicht zu erwarten ist, dass auch SAKP einen Inforamtionswert hat, der nicht null ist. Wir vermuten aber dass, das irgendwie zufällig ist oder irgendwelche zufällige Korrelation zwischen den vorhergesagten Werten und die interpolierten Werten von SAKP gefunden wurde. Auf jeden Fall ist dieses Ergebnis als Outlier zu betrachten.

12 Conclusion

Und so komme ich zu Schluss. Wir haben gesehen, in bestimmten Situationen können die gebauten Modellen die gestellte Ziele erreichen. Außerdem zeigen mehr oder weniger die Feature Importance Daten, das was wir wollten - etwas konsistentes über einige Modelle. Natürlich aber gibt's Möglichkeiten für Weiterentwicklung. Man kann au jeden Fall Modelle mit besserten Scores trainieren, wenn man sich mit den Hyperparametern spielt.

Ich bedanke mich bei Ihnen für die Aufmerksamkeit.

Contents