Plots

Kennzahlenvergleich XGboost/ARMA Aufgabe 1

Daten einlesen

```
library(ggplot2)
library(tidytext)
library(Metrics)
```

```
setwd("~/GitHub/fallstudien_2_projekt_1/prediction_results")

data_ul <- read.csv("predictions_ul.csv", header=TRUE, sep=",", dec=".")
data_dl <- read.csv("predictions_dl.csv", header=TRUE, sep=",", dec=".")
```

Erstelle Listen für die Kennzahlen der Modelle

```
kennzahlen_ul_xgboost <- list("vodafone" = list(),
                          "tmobile" = list(),
                          "o2" = list())

kennzahlen_ul_arima <- list("vodafone" = list(),
                            "tmobile" = list(),
                            "o2" = list())

kennzahlen_dl_xgboost <- list("vodafone" = list(),
                            "tmobile" = list(),
                            "o2" = list())

kennzahlen_dl_arima <- list("vodafone" = list(),
                        "tmobile" = list(),
                        "o2" = list())
```

Berechne den MAE und R^2

```
for (provider in c("vodafone", "tmobile", "o2")){
  subset_ul <- data_ul[data_ul$provider == provider, ]
  subset_dl <- data_dl[data_dl$provider == provider, ]

  kennzahlen_ul_xgboost[[provider]]$mae <- mae(subset_ul$prediction_xgboost,
                                               subset_ul$throughput_mbits)
  kennzahlen_ul_xgboost[[provider]]$rsquared <- 1 -
    sum((subset_ul$prediction_xgboost-subset_ul$throughput_mbits)^2)/
    sum((mean(subset_ul$throughput_mbits)-subset_ul$throughput_mbits)^2)
```

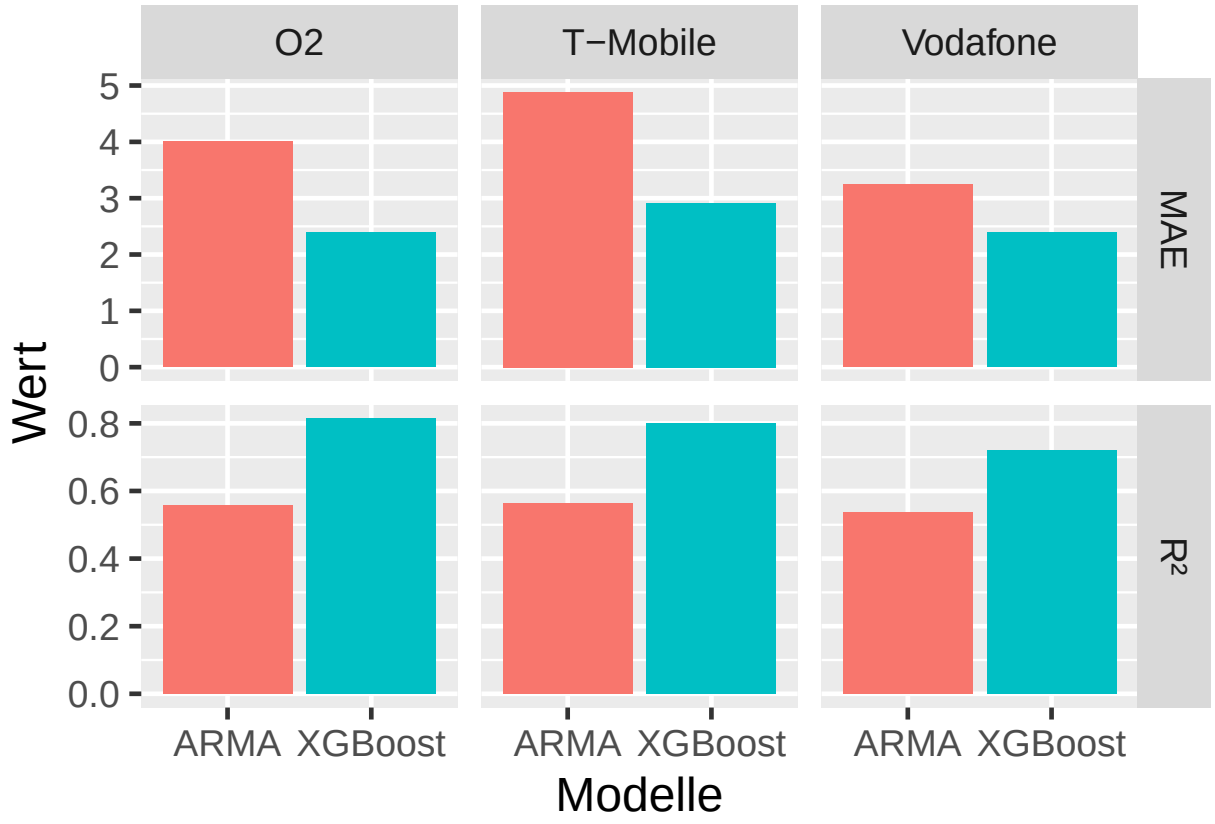
```
  kennzahlen_dl_xgboost[[provider]]$mae <- mae(subset_dl$prediction_xgboost,
                                               subset_dl$throughput_mbits)
  kennzahlen_dl_xgboost[[provider]]$rsquared <- 1 -
    sum((subset_dl$prediction_xgboost-subset_dl$throughput_mbits)^2)/
    sum((mean(subset_dl$throughput_mbits)-subset_dl$throughput_mbits)^2)

  kennzahlen_ul_arima[[provider]]$mae <- mae(subset_ul$prediction_arima,
                                             subset_ul$throughput_mbits)
  kennzahlen_ul_arima[[provider]]$rsquared <- 1 -
    sum((subset_ul$prediction_arima-subset_ul$throughput_mbits)^2)/
    sum((mean(subset_ul$throughput_mbits)-subset_ul$throughput_mbits)^2)

  kennzahlen_dl_arima[[provider]]$mae <- mae(subset_dl$prediction_arima,
                                             subset_dl$throughput_mbits)
  kennzahlen_dl_arima[[provider]]$rsquared <- 1 -
    sum((subset_dl$prediction_arima-subset_dl$throughput_mbits)^2)/
    sum((mean(subset_dl$throughput_mbits)-subset_dl$throughput_mbits)^2)

}
```

Uplink

```
df_ul <- data.frame(model = rep(c("XGBoost", "ARMA"), each = 6),
                    provider = rep(c("Vodafone", "T-Mobile", "O2"), 4),
                    kennzahl = c(rep(c("MAE", "R²"), each = 3),rep(c("MAE", "R²"), each = 3)),
                    value = c(kennzahlen_ul_xgboost$vodafone$mae,
                              kennzahlen_ul_xgboost$tmobile$mae,
                              kennzahlen_ul_xgboost$o2$mae,
                              kennzahlen_ul_xgboost$vodafone$rsquared,
                              kennzahlen_ul_xgboost$tmobile$rsquared,
                              kennzahlen_ul_xgboost$o2$rsquared,
                              kennzahlen_ul_arima$vodafone$mae,
                              kennzahlen_ul_arima$tmobile$mae,
                              kennzahlen_ul_arima$o2$mae,
                              kennzahlen_ul_arima$vodafone$rsquared,
                              kennzahlen_ul_arima$tmobile$rsquared,
                              kennzahlen_ul_arima$o2$rsquared))
```

Histogrammplot der Kennzahlen im Uplink

```
ggplot(data = df_ul, aes(x = model, y = value, fill = model) )+
  geom_bar(stat = "identity", position = position_dodge()) +
  facet_grid(kennzahl ~ provider, scales = "free_y") +
  theme_grey(base_size = 18) +
  theme(legend.title = element_blank(),
        legend.position = "none") +
  xlab("Modelle") +
  ylab("Wert")
```



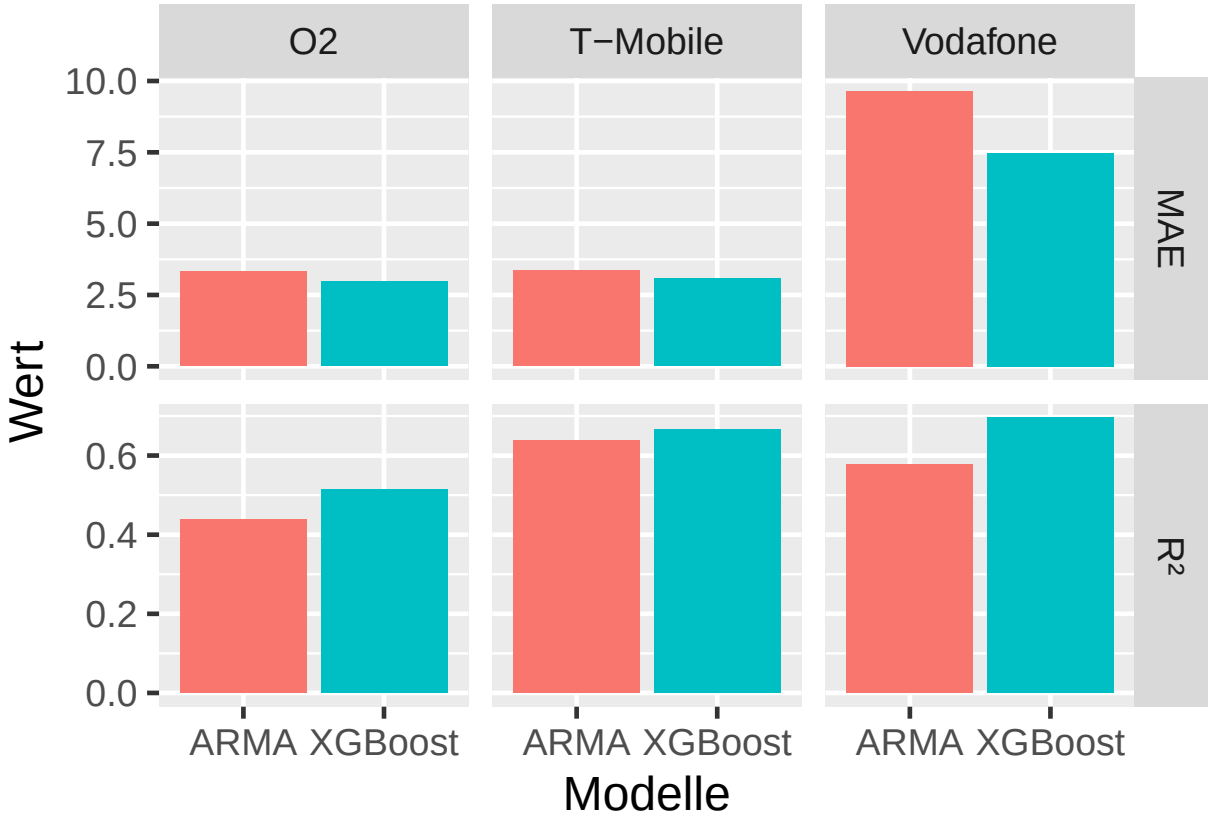
Downlink

```
df_dl <- data.frame(model = rep(c("XGBoost", "ARMA"), each = 6),
                    provider = rep(c("Vodafone", "T-Mobile", "O2"), 4),
                    kennzahl = c(rep(c("MAE", "R²"), each = 3),rep(c("MAE", "R²"), each = 3)),
                    value = c(kennzahlen_dl_xgboost$vodafone$mae,
                              kennzahlen_dl_xgboost$tmobile$mae,
                              kennzahlen_dl_xgboost$o2$mae,
                              kennzahlen_dl_xgboost$vodafone$rsquared,
                              kennzahlen_dl_xgboost$tmobile$rsquared,
                              kennzahlen_dl_xgboost$o2$rsquared,
                              kennzahlen_dl_arima$vodafone$mae,
                              kennzahlen_dl_arima$tmobile$mae,
                              kennzahlen_dl_arima$o2$mae,
                              kennzahlen_dl_arima$vodafone$rsquared,
                              kennzahlen_dl_arima$tmobile$rsquared,
                              kennzahlen_dl_arima$o2$rsquared))
```

Histogrammplot der Kennzahlen im Downlink

```
ggplot(data = df_dl, aes(x = model, y = value, fill = model) )+
  geom_bar(stat = "identity", position = position_dodge()) +
  facet_grid(kennzahl ~ provider, scales = "free_y") +
```

```
  theme_grey(base_size = 18) +
  theme(legend.title = element_blank(),
        legend.position = "none") +
  xlab("Modelle") +
  ylab("Wert")
```



Feature Importance Aufgabe 2

Daten einlesen

```
setwd("~/GitHub/fallstudien_2_projekt_1/prediction_results")

data <- read.csv("feature_importance_xgboost_linklifetime.csv", header = TRUE)
```

```
df_ll <- data.frame(provider = rep(c(" ", "  ", "   "), each = 9),
                    #features = data$feature[-which(c(data$feature == "enodeb"))],
                    features = data$feature,
                    #value = data$Gain[-which(c(data$feature == "enodeb"))])
                    value = abs(data$Permutation))
```

Histogramm Plot der Feature Importance der verschiedenen Provider

```r
name_mapping = c(
  " " = "O2",
  "  " = "T-Mobile",
  "   " = "Vodafone"
)

ggplot(data = df_ll, aes(x = reorder_within(features, -value, provider, sep = " "),
                         y = value, fill = provider)) +
  geom_bar(stat = "identity" ) +
  facet_wrap(~ provider, scales = "free", labeller = as_labeller(name_mapping)) +
  theme_grey(base_size = 18) +
  theme(legend.title = element_blank(), axis.text.x = element_text(angle = -45, hjust = 0,
                                                                   vjust = 0.5),
        legend.position = "none") +
  xlab("Features") +
  ylab("Koeffizienten")
```