Interpretable Machine Learning

Interpretable Models

Learning goals

- Examples for interpretable models: (generalized) linear models, generalized additive models, model-based boosting

LINEAR REGRESSION

- For linear regression models, we only estimate the model parameters. The model equation is manually specified and known in advance:

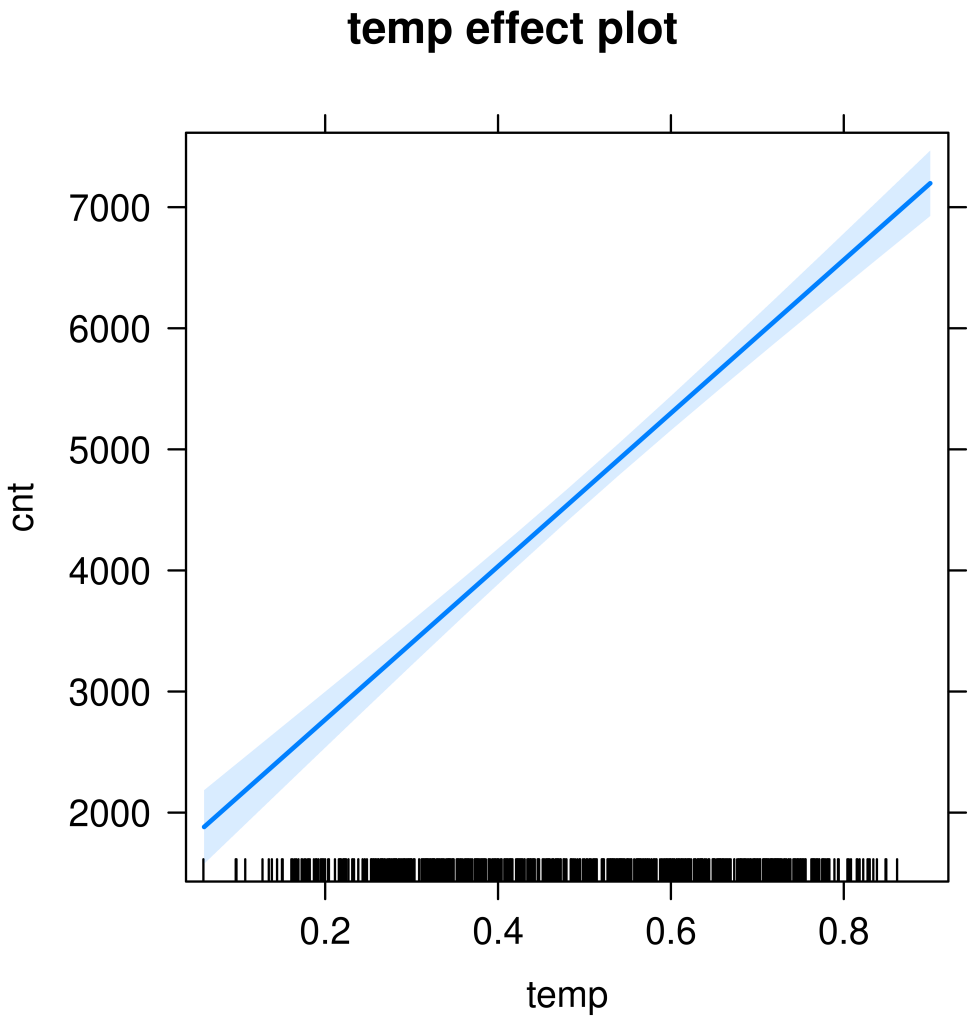
$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \cdots + \epsilon$$

- The predictive power of LMs is determined by specifying the correct model structure.
- The model equation is identical across the entire feature space, i.e., local interpretations are identical to global ones. By knowing the model equation, we can exactly determine feature effects (e.g., beta coefficients, effect plots) and importance scores (e.g., p-values, t-statistics).
- Note that for inference-based metrics (p-values, t-statistics, confidence intervals) to be valid, the error term needs to be normally distributed with zero mean (i.e., $(Y|X) \sim N(x^T\beta, \sigma^2)$. This severely restricts the usage of LMs in practice.

LINEAR REGRESSION

```
Call:
lm(formula = cnt ~ (hum + temp + windspeed + weathersit)^2, data = data_bike)

Residuals:
    Min      1Q  Median      3Q     Max
-4231.4 -1054.7  -119.3  1035.1  3673.1

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)            1204.5     1344.8   0.896   0.3707
hum                    3243.9     2224.3   1.458   0.1452
temp                   7187.4     1764.3   4.074 5.14e-05 ***
windspeed             -1572.8     3209.8  -0.490   0.6243
weathersit              476.8      639.7   0.745   0.4563
hum:temp              -5873.1     2880.3  -2.039   0.0418 *
hum:windspeed          1552.6     5601.7   0.277   0.7817
hum:weathersit        -1698.9      668.1  -2.543   0.0112 *
temp:windspeed         2788.2     4416.4   0.631   0.5280
temp:weathersit        1646.3      730.8   2.253   0.0246 *
windspeed:weathersit  -3017.8     1563.4  -1.930   0.0540 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1401 on 720 degrees of freedom
Multiple R-squared:  0.484,Adjusted R-squared:  0.4768
F-statistic: 67.53 on 10 and 720 DF,  p-value: < 2.2e-16
```

LINEAR REGRESSION



GENERALIZED LINEAR REGRESSION

- Generalized linear models (GLMs) are able to model data more flexibly through the link function, but keep the pre-specified model equation:

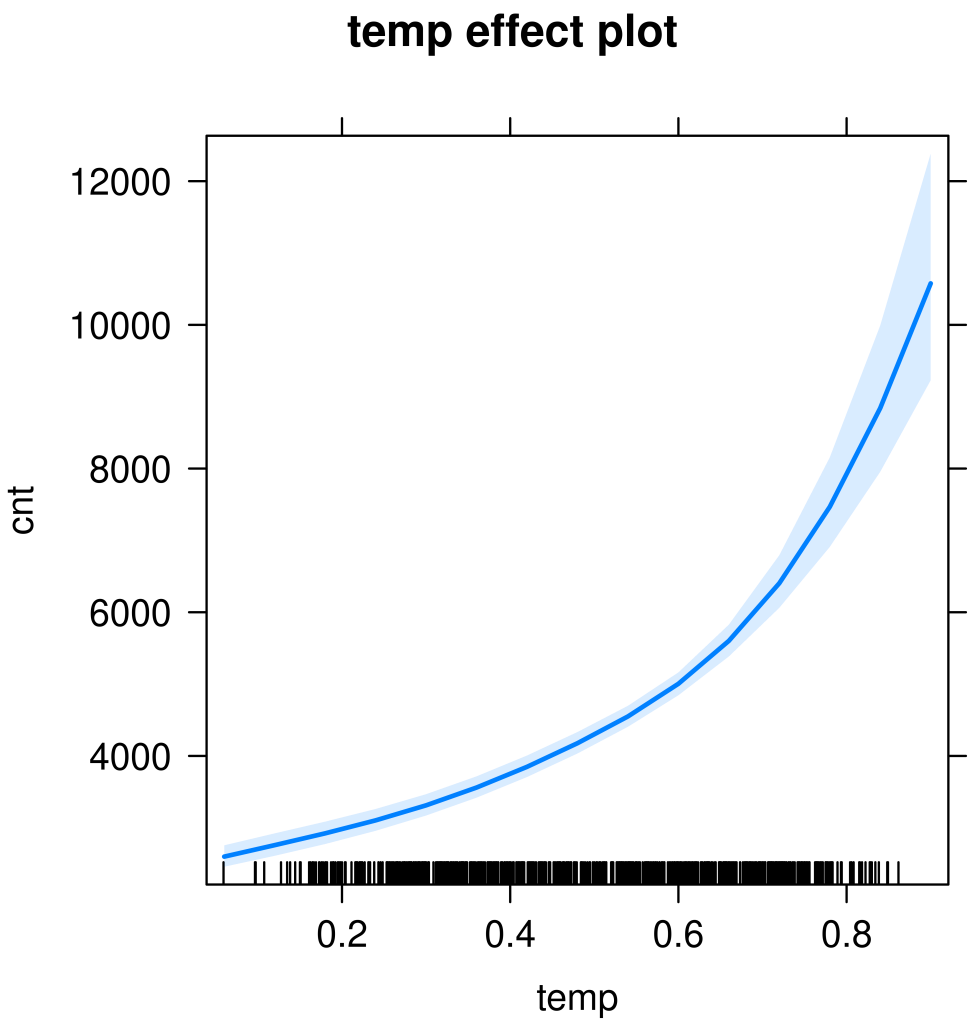
$$g\left(\mathbb{E}_Y(y|x)\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p$$

- This flexibility relates to the distribution of the target. The target conditional on the features can follow a distribution of the exponential family, e.g., Binomial, Poisson, Exponential, Gamma. We still need to specify the correct model equation in advance in order to receive a good model fit.
- As the model equation is still known, interpretations are possible the same way as for linear models. However, even linear terms become non-linear through the link function!

GENERALIZED LINEAR REGRESSION

```
Call:
glm(formula = cnt ~ (hum + temp + windspeed + weathersit)^2,
    family = Gamma(link = "inverse"), data = data_bike)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.61704  -0.28690  -0.02012   0.21600   0.87973

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           2.965e-04  9.480e-05   3.128 0.001834 **
hum                  -2.486e-04  1.490e-04  -1.669 0.095612 .
temp                  7.827e-05  1.120e-04   0.699 0.485014
windspeed             1.634e-04  2.255e-04   0.724 0.469049
weathersit           -5.377e-06  4.472e-05  -0.120 0.904327
hum:temp              1.090e-04  1.720e-04   0.634 0.526266
hum:windspeed         2.741e-04  3.415e-04   0.803 0.422490
hum:weathersit        1.477e-04  4.522e-05   3.265 0.001145 **
temp:windspeed       -1.015e-03  2.706e-04  -3.751 0.000191 ***
temp:weathersit      -2.143e-04  4.352e-05  -4.923 1.06e-06 ***
windspeed:weathersit  2.988e-04  1.049e-04   2.849 0.004514 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Gamma family taken to be 0.1345313)

    Null deviance: 189.26  on 730  degrees of freedom
Residual deviance: 116.36  on 720  degrees of freedom
AIC: 12885
```

GENERALIZED LINEAR REGRESSION



GENERALIZED ADDITIVE MODELS

- A generalized additive model (GAM) adds flexibility and predictive power by replacing pre-specified terms with smoothing functions:

$$g\left(\mathbb{E}_Y(y|x)\right) = \beta_0 + \beta_1 h_1(x_1) + \cdots + \beta_p h_p(x_p) + \ldots$$

- For the component functions, we may either specify a parametric form (e.g., a regression splines), or a non-parametric one, e.g., locally estimated scatterplot smoothing (LOESS).
- This makes GAMs much more flexible regarding the model structure, which is largely determined by the structure of the data instead of premade assumptions as in LMs and GLMs.
- A GAM retains interpretability by keeping the additive model equation (as long as the component functions are interpretable). As the model equation is known, we can use similar interpretation methods as for LMs and GLMs, e.g., evaluating the estimated functions that depend on the features of interest.

INTERPRETABLE MODELS

Model-based boosting:

- Idea: Combine boosting with interpretable base learners (e.g., linear model with single parameter).
- Consider two linear base learners $b_j(x, \Theta)$ and $b_j(x, \Theta^\star)$ with the same type, but distinct parameter vectors Θ and $\Theta^\star$. They can be combined in a base learner of the same type:

$$b_j(x, \Theta) + b_j(x, \Theta^\star) = b_j(x, \Theta + \Theta^\star)$$

- We create a selection of interpretable base learners. In each iteration, all base learners are trained on the so-called pseudo residuals, and the one with the best fit is added to the previously computed model.
- The final model has an additive structure (equivalent to a GAM), where each component function is itself interpretable.

INTERPRETABLE MODELS

Rule-based ML:
Decision rules follow a general structure: IF the conditions are met THEN make a certain prediction. A single decision rule or a combination of several rules can be used to make predictions.

There are many ways to learn rules from data:

- OneR learns rules from a single feature. OneR is characterized by its simplicity, interpretability and its use as a benchmark.
- Sequential covering is a general procedure that iteratively learns rules and removes the data points that are covered by the new rule. This procedure is used by many rule learning algorithms.
- Bayesian Rule Lists combine pre-mined frequent patterns into a decision list using Bayesian statistics. Using pre-mined patterns is a common approach used by many rule learning algorithms.