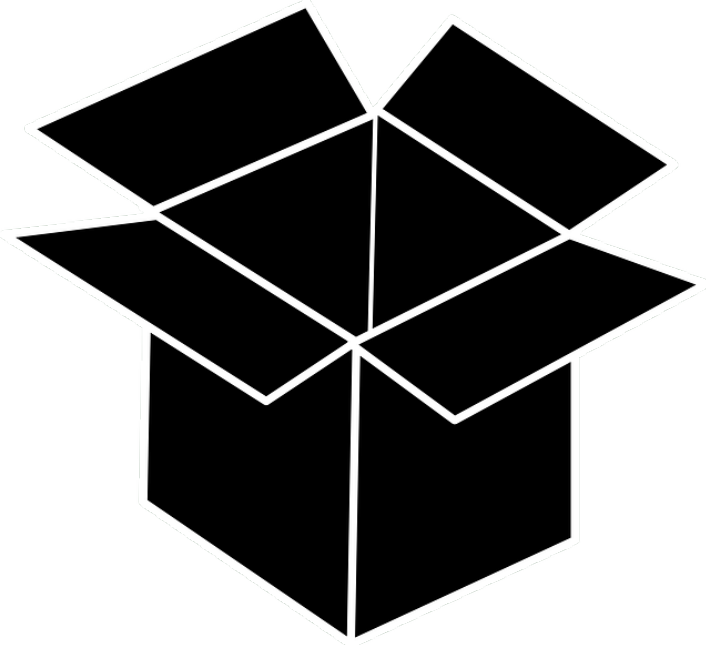
Interpretable Machine Learning

Interpretable Models



Learning goals

- Examples for interpretable models: linear and polynomial regression models, generalized linear models, generalized additive models, model-based boosting, rule-based learning
- Methodological summary, motivation behind the model, model interpretations

LINEAR AND POLYNOMIAL REGRESSION

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

$$y = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p + \cdots + \epsilon$$
$$Y = X^T \beta + \mathcal{E}$$

- The model equation is identical across the entire feature space.

- The predictive power of LMs is determined by specifying the correct model structure. A polynomial regression model is an extension of the LM that includes higher order terms or interactions. This way we can additively model non-linear data and make use of the entire LM toolbox.

LINEAR AND POLYNOMIAL REGRESSION

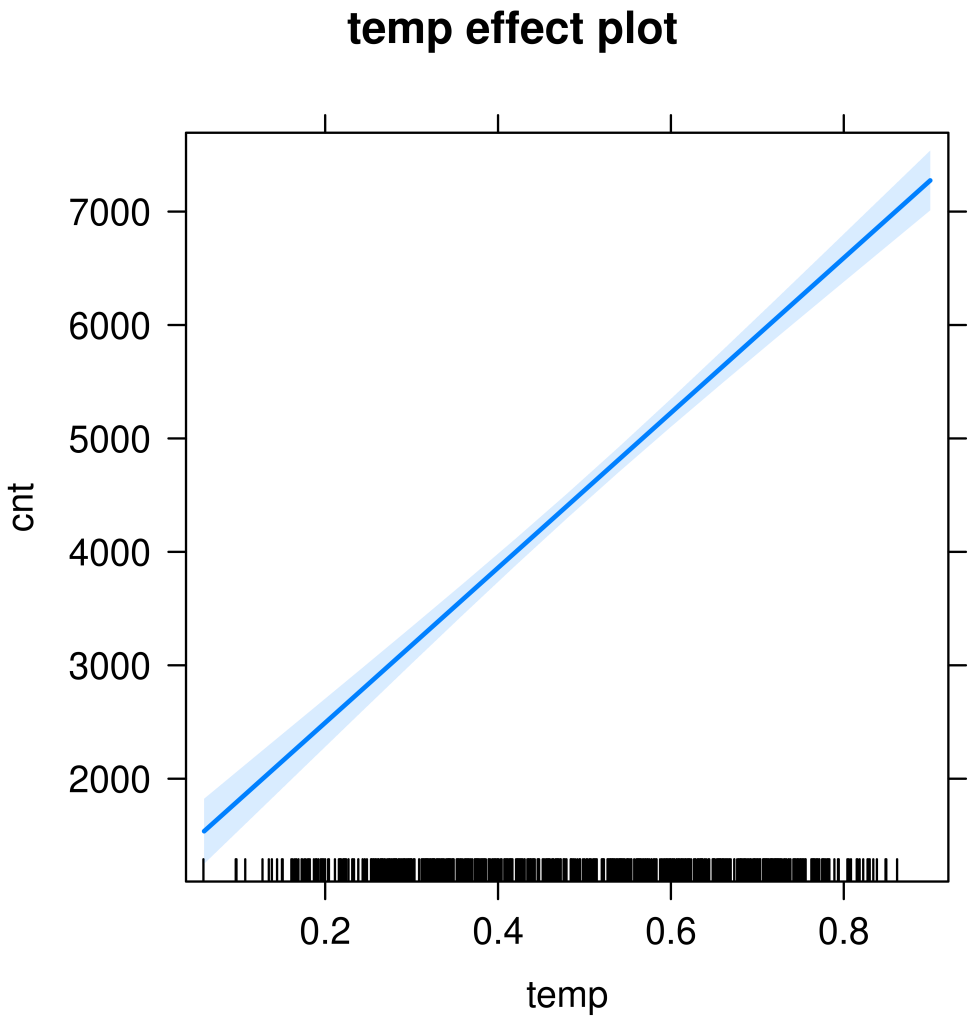
- By knowing the model equation, we can exactly determine feature effects (e.g., beta coefficients, effect plots) and importance scores (e.g., p-values, t-statistics). For higher order effects or interactions, beta coefficients cannot be interpreted in isolation, i.e., we need to use marginal effects or effect plots with simultaenous changes in feature values.

- 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., $\epsilon \sim N(0, \sigma^2)$. It follows that $(y|x) \sim N(x^T\beta, \sigma^2)$. This restricts the usage of LMs in practice, as the distribution of the error term is a prior assumption about the data.

LINEAR AND POLYNOMIAL REGRESSION MODELS

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

Residuals:
    Min      1Q  Median      3Q     Max
-4635.2 -1122.0   -82.6  1017.2  3528.9

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2112.4      695.5   3.037  0.00247 **
hum          -1568.5     1150.5  -1.363  0.17320
temp          8085.1     1437.0   5.627 2.63e-08 ***
hum:temp     -1998.9     2344.8  -0.852  0.39423
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1469 on 727 degrees of freedom
Multiple R-squared:  0.4274,Adjusted R-squared:  0.425
F-statistic: 180.9 on 3 and 727 DF,  p-value: < 2.2e-16
```

LINEAR AND POLYNOMIAL REGRESSION MODELS



GENERALIZED LINEAR REGRESSION MODELS

- Generalized linear models (GLMs) are more flexible regarding the target distribution. They keep the linear predictor $X^T\beta$ which now explains the transformed, expected conditional target through a link function g:

$$g\left(\mathbb{E}_Y(Y|X)\right) = X^T\beta$$
$$\mathbb{E}_Y\left(Y|X\right) = g^{-1}(X^T\beta)$$

- GLMs are a framework for target distributions of the exponential family, e.g., Gaussian, Binomial, Poisson, Exponential, Gamma. A Gaussian target distribution with identity link corresponds to a linear / polynomial regression model.

GENERALIZED LINEAR REGRESSION MODELS

- The link function describes how the linear predictor $X^T\beta$ relates to the expected, conditional target $\mathbb{E}_Y(Y|X)$, e.g., if the target is distributed binomially, a natural link function is the logit link $log\left(\frac{\mathbb{E}_Y(Y|X)}{1-\mathbb{E}_Y(Y|X)}\right)$.

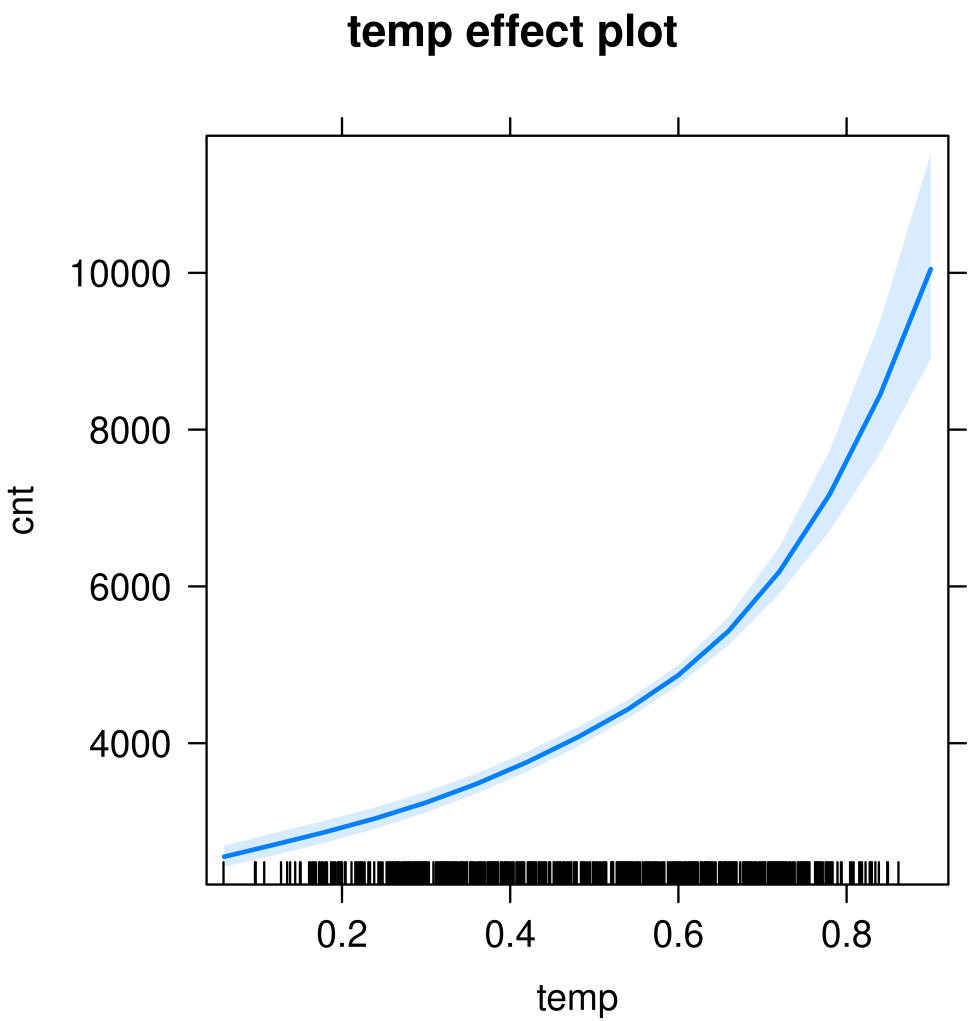
- We need to specify the correct model equation, target distribution, and link function in order to receive a good model fit.

- As the model equation is still known, interpretations are possible the same way as for polynomial regression models. However, even linear terms become non-linear through the link function (if the identity link is not used)!

GENERALIZED LINEAR REGRESSION MODELS

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

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.84829  -0.31351  -0.01372   0.22187   0.75445

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.618e-04  4.648e-05   5.633 2.54e-08 ***
hum          2.404e-04  7.902e-05   3.042  0.00243 **
temp        -1.846e-04  8.243e-05  -2.239  0.02545 *
hum:temp    -2.603e-04  1.400e-04  -1.859  0.06342 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Gamma family taken to be 0.1454719)

    Null deviance: 189.26  on 730  degrees of freedom
Residual deviance: 132.01  on 727  degrees of freedom
AIC: 12966

Number of Fisher Scoring iterations: 5
```

GENERALIZED LINEAR REGRESSION MODELS



GENERALIZED ADDITIVE MODELS

- A generalized additive model (GAM) adds flexibility and predictive power to the GLM framework 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 adaptive. The estimated model is largely determined by the structure of the data instead of premade assumptions as in LMs and GLMs. However, we still need to specify the order and type of the component functions to be estimated, e.g., which interactions to include.

GENERALIZED ADDITIVE MODELS

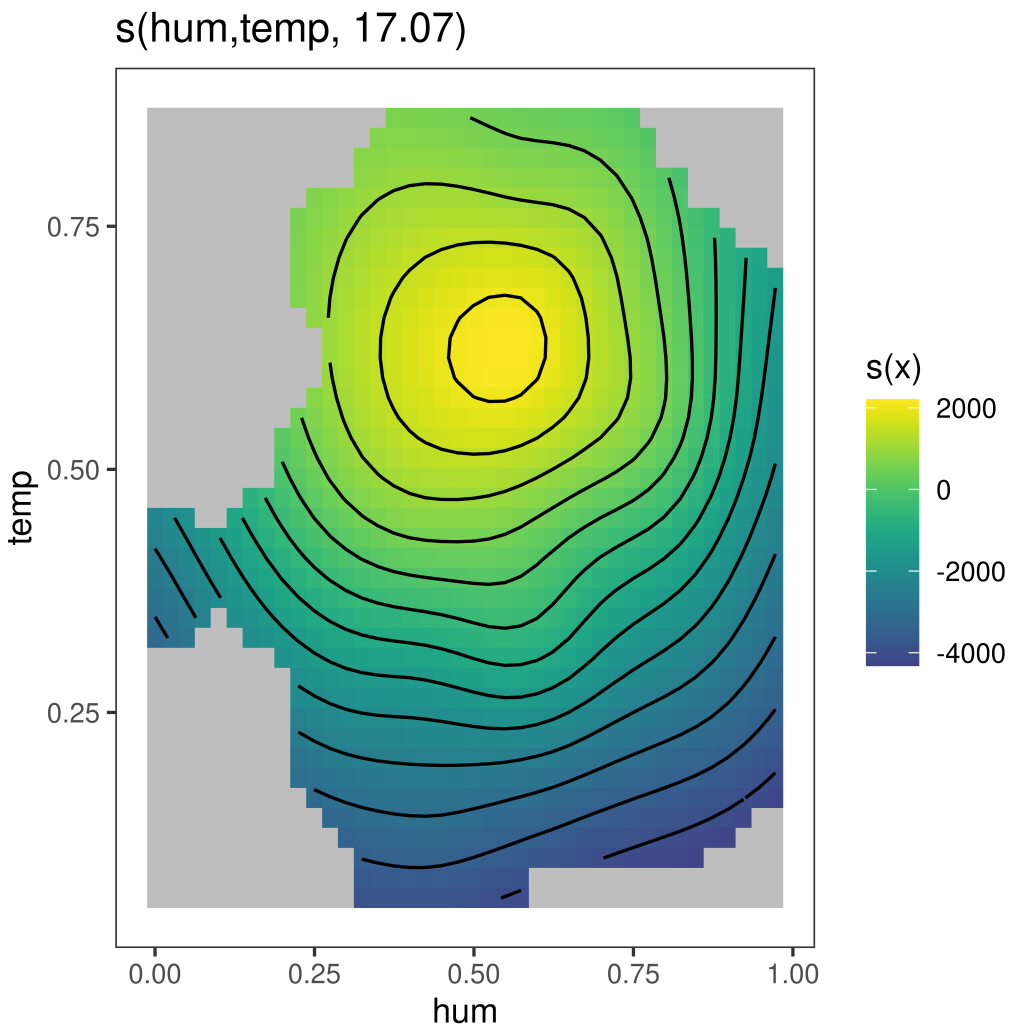
- Furthermore, the smoothing degree of each component function can be tuned to avoid overfitting. Regularization increases interpretability, as the interpretations drawn from the model can be transferred more easily to new data.

- 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 components that depend on the features of interest.

GENERALIZED ADDITIVE MODELS

```
Family: gaussian
Link function: identity

Formula:
cnt ~ s(hum, temp)

Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4504.35      46.83   96.18   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Approximate significance of smooth terms:
               edf Ref.df     F p-value
s(hum,temp) 17.07   22.1 44.09  <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

R-sq.(adj) =  0.573   Deviance explained = 58.3%
GCV = 1.6439e+06  Scale est. = 1.6033e+06  n = 731
```

GENERALIZED ADDITIVE MODELS



MODEL-BASED BOOSTING

- Boosting iteratively joins weak base learners to create a powerful ensemble model.

- Idea: Combine boosting with interpretable base learners (e.g., single feature LM). The resulting ensemble is also interpretable.

- 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:

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

MODEL-BASED BOOSTING

- 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:

$$\widehat{f}^{[1]}(x) = f_0 + \beta b_3(x_3, \theta^{[1]})$$
$$\widehat{f}^{[2]}(x) = f_0 + \beta b_3(x_3, \theta^{[1]}) + \beta b_3(x_3, \theta^{[2]})$$
$$\widehat{f}^{[3]}(x) = f_0 + \beta b_3(x_3, \theta^{[1]}) + \beta b_3(x_3, \theta^{[2]}) + \beta b_1(x_1, \theta^{[3]})$$

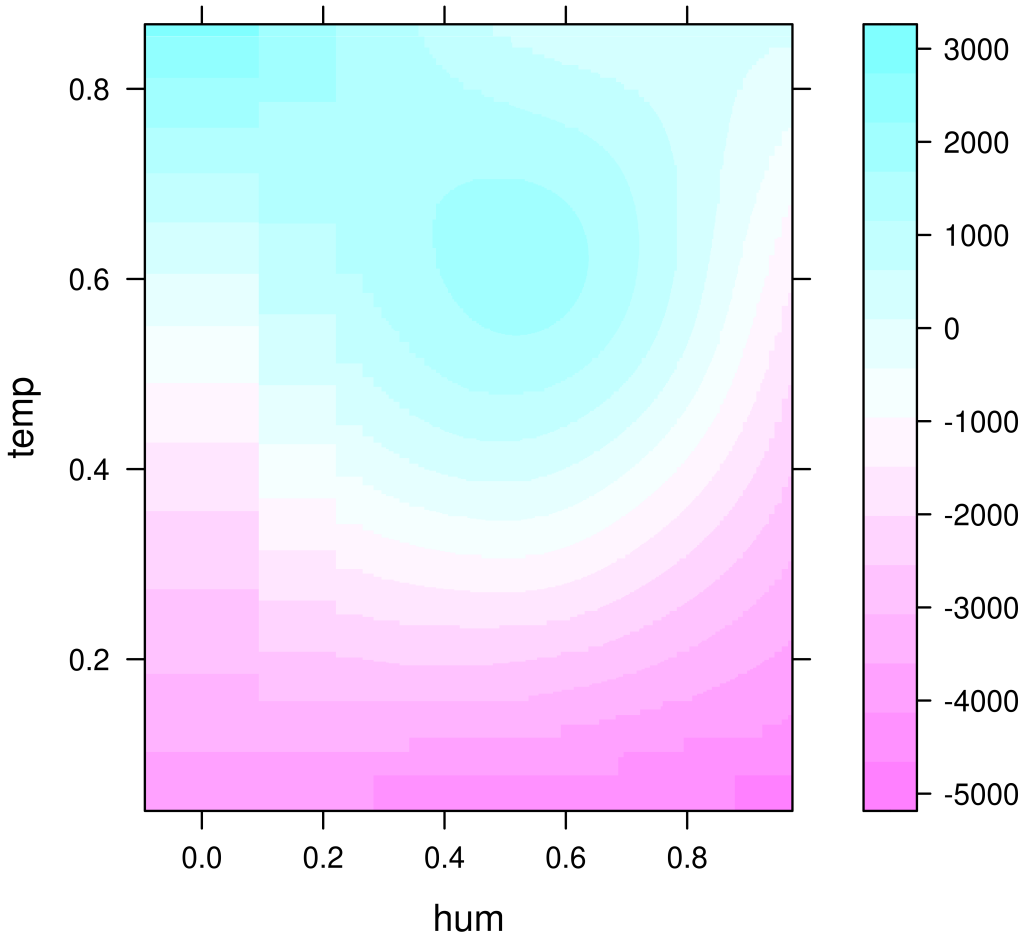
$$\widehat{f}^{[3]}(x) = f_0 + \beta \left(b_3(x_3, \theta^{[1]} + \theta^{[2]}) + b_1(x_1, \theta^{[3]}) \right)$$

- The final model has an additive structure (equivalent to a GAM), where each component function is interpretable itself.

MODEL-BASED BOOSTING

```
 Model-based Boosting

Call:
mboost(formula = cnt ~ bols(hum) + bols(temp) + bspatial(hum,     temp), data = data_bike)


 Squared Error (Regression)

Loss function: (y - f)^2


Number of boosting iterations: mstop = 100
Step size:  0.1
Offset:  4504.349
Number of baselearners:  3

Selection frequencies:
bspatial(hum, temp)
                  1
```

MODEL-BASED BOOSTING



RULE-BASED LEARNING

- Decision rules follow a general structure: IF the conditions are met, THEN make a certain prediction. Rule-based learning aims to capture a set of decision rules that accurately describe the data.

- Although a single decision rule is perhaps the most interpretable model, a large collection of decision rules might not be.

- One needs to differentiate between descriptive rule discovery (DRD) and predictive rule learning (PRL). DRD is based on mining databases for frequent item sets, and converting these into a set of decision rules, e.g., a decision rule {mobile phone} $\Rightarrow$ {mobile phone case} might stem from mining a frequent pattern of mobile phones and phone cases being bought together. DRD is not suited to make predictions for unseen data. Conversely, PRL aims to

RULE-BASED LEARNING

discover a set of rules that covers the entire feature space, thereby being able to predict for any data instance.

- There are innumerable rule-based learning algorithms. Decision tree learning is the most popular group of rule-based models, as every tree can be represented as a collection of decision rules (and subsequent tree splits as a series of rules). Other variants include OneR, Sequential Covering, or Bayesian Rule Lists.

Fürnkranz J., Kliegr T. (2015) A Brief Overview of Rule Learning. In: Bassiliades N., Gottlob G., Sadri F., Paschke A., Roman D. (eds) Rule Technologies: Foundations, Tools, and Applications. RuleML 2015. Lecture Notes in Computer Science, vol 9202. Springer, Cham.