STAT 380:
Variable Selection, Ridge, and Lasso Regression

An Example using .... Data

United Arab Emirates University

Let $\{y_i, \mathbf{X}_i\}_{i=1}^{n}$ be the observed data. And there is a statistical/machine learning model that provides a prediction for the responses y_i.

We denoted the predicted value for the responses to be $\hat{y}_i$

The Boston Housing Dataset

Boston Housing data set

Boston Housing data set

- The Boston Housing data contain information on census tracts in suburbs of Boston.
- Several measurements are included (e.g., crime rate, pupil-teacher ratio).
- 14 variables for each of the 506 houses.

Possible tasks:

- A supervised predictive task, where the outcome is the median value of a home.
- A supervised classification task, where the outcome is the binary variable *CAT.MEDV* that indicates whether the home value is above or below $\$30,000$.
- An unsupervised task, where the goal is to cluster houses.

Variables in the Boston housing data set

Variable	Name
Crime rate	CRIM
Percentage of residential land zoned for lots over 25,000 ft^2	ZN
Percentage of land occupied by nonretail business	INDUS
Does tract bound Charles River (= 1 if tract bounds river)	CHAS
Nitric oxide concentration (parts per 10 million)	NOX
Average number of rooms per dwelling	RM
Percentage of owner-occupied units built prior to 1940	AGE
Weighted distances to five Boston employment centers	DIS
Index of accessibility to radial highways	RAD
Full-value property tax rate per $10,000	TAX
Pupil-to-teacher ratio by town	PTRATIO
Percentage of lower status of the population	LSTAT
Median value of owner-occupied homes in $1000s	MEDV
Is median value of owner-occupied homes in tract above $30,000 (CAT.MEDV = 1) or not (CAT.MEDV = 0)	CAT.MEDV

Boston housing data set: Overview

The head()-command returns the first parts of a vector, matrix, table, data frame or function.

```
> head(Daten)
 CRIM ZN INDUS   RM  AGE   DIS RAD TAX LSTAT MEDV CMEDV
0.006 18  2.31 6.57 65.2 4.090   1 296  4.98 24.0     0
0.027  0  7.07 6.42 78.9 4.967   2 242  9.14 21.6     0
0.027  0  7.07 7.18 61.1 4.967   2 242  4.03 34.7     1
0.032  0  2.18 6.99 45.8 6.062   3 222  2.94 33.4     1
0.069  0  2.18 7.14 54.2 6.062   3 222  5.33 36.2     1
0.029  0  2.18 6.43 58.7 6.062   3 222  5.21 28.7     0
```

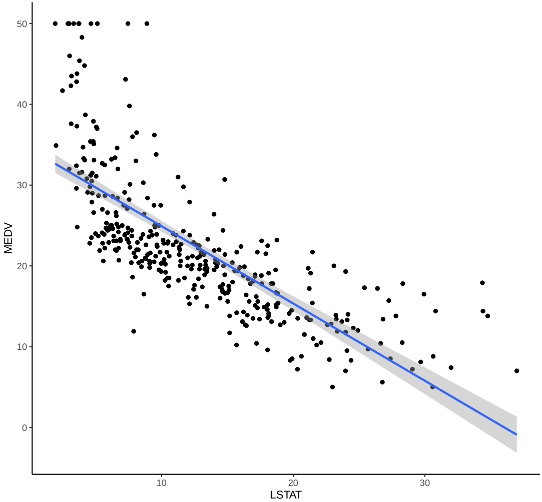
Boston housing data set: Overview

The `str()`-command displays the internal structure of an R object.

```
> str(Daten)
'data.frame': 506 obs. of  14 variables:
 $ CRIM      : num  0.00632 0.02731 0.02729 0.03237 ...
 $ ZN        : num  18 0 0 0 0 0 12.5 12.5 12.5 12.5 ...
 $ INDUS     : num  2.31 7.07 7.07 2.18 2.18 2.18 7.87 ...
 $ CHAS      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ NOX       : num  0.538 0.469 0.469 0.458 0.458 ...
 $ RM        : num  6.58 6.42 7.18 7 7.15 ...
 $ AGE       : num  65.2 78.9 61.1 66.6 96.1 100 85.9 ...
 $ DIS       : num  4.09 4.97 4.97 6.06 6.06 ...
 $ RAD       : int  1 2 2 3 3 3 5 5 5 5 ...
 $ TAX       : int  296 242 242 311 311 311 ...
 $ PTRATIO   : num  15.3 17.8 17.8 18.7 18.7 18.7 15.2 ...
 $ LSTAT     : num  4.98 9.14 4.03 2.94 5.33 ...
 $ MEDV      : num  24 21.6 34.7 33.4 36.2 28.7  ...
 $ CAT..MEDV: int  0 0 1 1 1 0 0 0 0 0 ...
```

Polynomial Regression

Illustration: Motivation



Polynomial regression

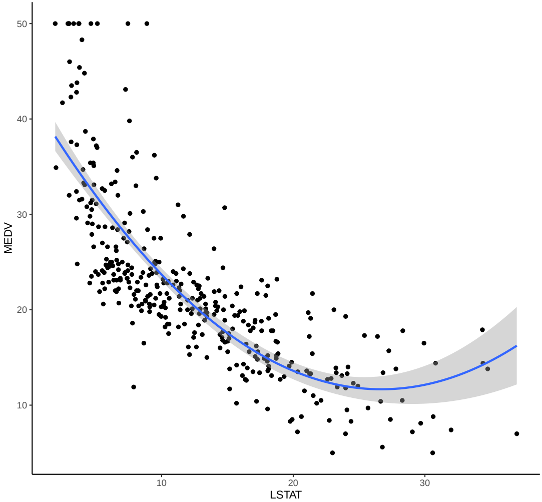
- It **extends** the linear model by adding extra predictors, obtained by raising each of the original predictors to a power.
- For example, a cubic regression uses three variables, X, X^2, and X^3, as predictors.
- This approach provides a simple way to provide a **nonlinear** fit to data.
- It is considered to be a special case of multiple linear regression.
- A polynomial regression may lead to increase in **complexity** as the number of covariates also increases.
- Polynomial models should be hierarchical, containing the terms X, X^2, and X^3, in a hierarchy.

Polynomial regression in R: Boston housing data

We can use the `poly()`-command for specifying a polynomial regression:

```
# To fit a polynomial model
modelfinal <- lm(MEDV ~ poly(LSTAT, 2, raw = TRUE), data
    = train)
# Make predictions
predictions <- modelfinal %>% predict(test)
# Model performance
data.frame(RMSE = RMSE(predictions, test$MEDV),
R2 = R2(predictions, test$MEDV))
# Let's check the curve
ggplot(train, aes(LSTAT, MEDV) ) + geom_point() +
stat_smooth(method = lm, formula = y ~ poly(x, 2, raw =
    TRUE))
# Let's check the assumptions
residuals <- data.frame('Residuals' = modelfinal$
    residuals)
```

Polynomial regression: Boston housing data



Regression Splines

Thank You