B4M36SAN
BE4M36SAN
Seminar/Tutorial

Alikhan Anuarbekov

Ali

anuarali@fel.cvut.cz

Big team project (Final assignment section in courseware**)**

- Start looking for 4 people team **NOW**

- Try to find a question/dataset that is interesting for you

- Look for previous year best projects for inspiration

- Ask question before the deadline for team creation (3.11)

- If no team on 3.11 (midterm week), then I will forcefully assign a team for you

Tips and recommendations:

- Do not focus on single question, try to combine two or more questions
  - Example: Not only look for medical question, but try to add socio-economy

- Try to find more than one dataset
  - Example: UNICEF + World Bank

- Start working early, at least to formulate a question

Practical assignment: compute the adjusted p-value

We have 5 variables: Y ~ X1, X2, X3, X4, X5

- Each variable has unknown alpha threshold
- We need **total_alpha (total Type I error) = 0.10**

$$1 - (1 - \alpha)^n$$

The method proposed by Šidák is defined as $p_j^{\check{\text{Si}}} = 1 - \left(1 - p_j\right)^M$. Equivalently, the significance level could be adjusted to $\alpha^{\check{\text{Si}}} = 1 - (1 - \alpha)^{1/M}$, where α is the unadjusted significance level. Under the assumption that the outcomes are independent, the adjustment can be derived as

$$P\left(no\; Type\; I\; error\; on\; \mathbf{1}\; test\right) = 1 - \alpha^{\check{\text{Si}}},$$

$$\rightarrow P\left(no\; Type\; I\; error\; on\; \mathbf{M}\; tests\right) = \left(1 - \alpha^{\check{\text{Si}}}\right)^M,$$

$$\rightarrow P\left(\boldsymbol{atleast}\; one\; Type\; I\; error\; on\; \mathrm{M}\; tests\right) = 1 - \left(1 - \alpha^{\check{\text{Si}}}\right)^M = \alpha.$$

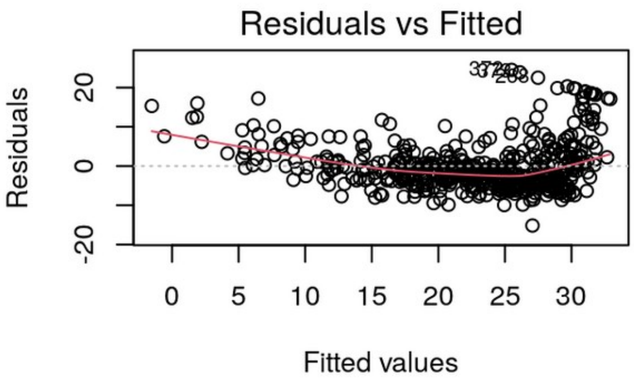
	Null hypothesis is true (H_0)	Alternative hypothesis is true (H_A)	Total
Test is declared significant	V	S	R
Test is declared non-significant	U	T	$m-R$
Total	m_0	$m-m_0$	m

- m is the total number hypotheses tested
- m_0 is the number of true null hypotheses, an unknown parameter
- $m-m_0$ is the number of true alternative hypotheses
- V is the number of false positives (Type I error) (also called "false discoveries")
- S is the number of true positives (also called "true discoveries")
- T is the number of false negatives (Type II error)
- U is the number of true negatives
- $R=V+S$ is the number of rejected null hypotheses (also called "discoveries", either true or false)

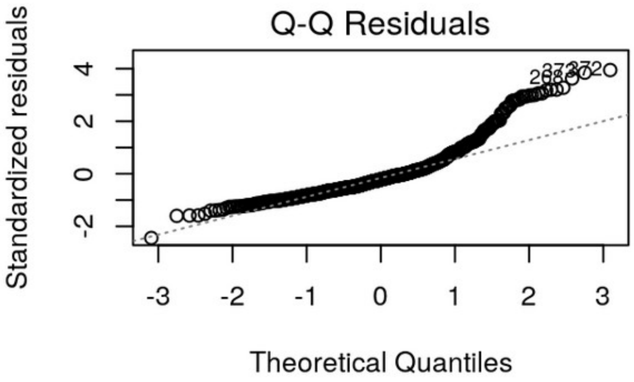
Now with the new topic

https://library.virginia.edu/data/articles/diagnostic-plots

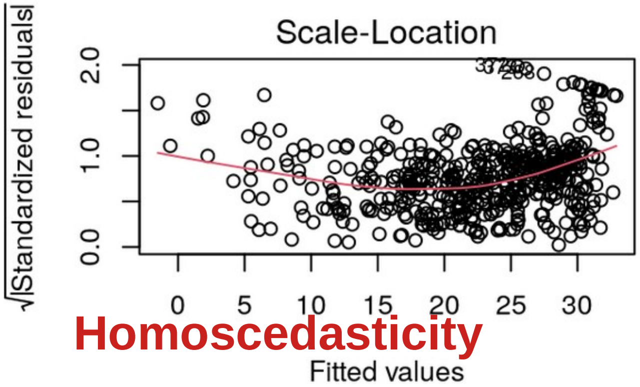
How to analyze the problems of linear regression visually via diagnostic plots



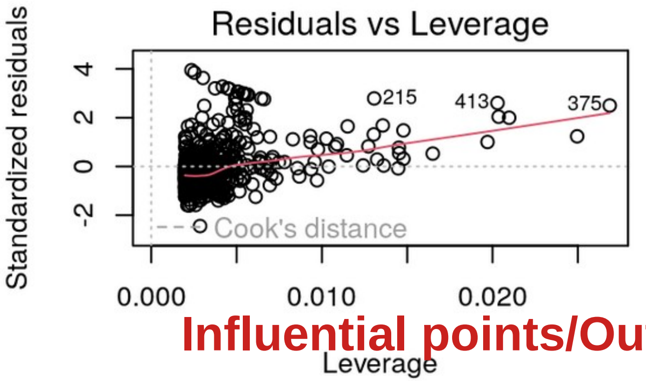
Linearity



Normality



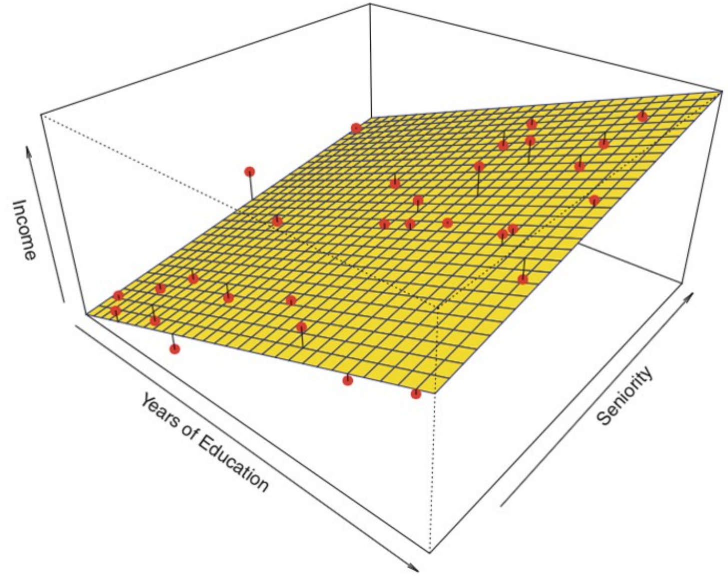
Homoscedasticity

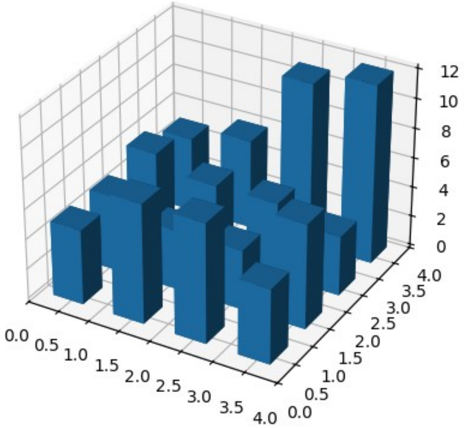


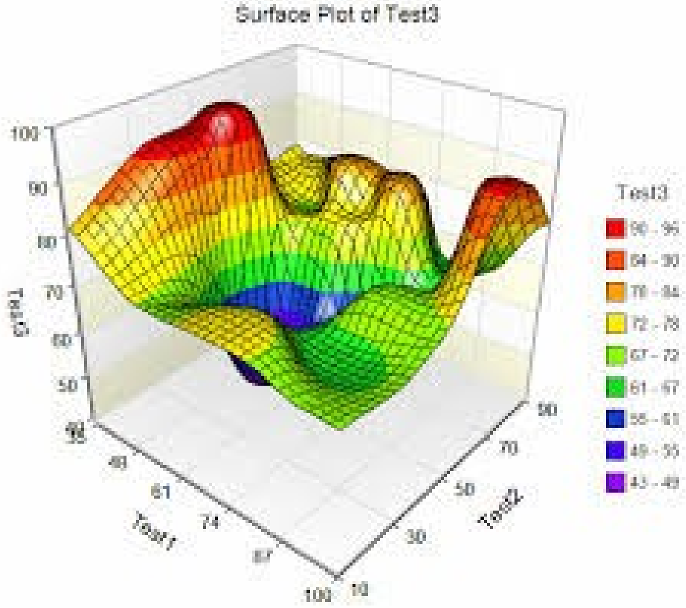
Influential points/Outliers?

$$Income = \beta_0 + \beta_1 \cdot Years + \beta_2 \cdot Seniority + \epsilon, \qquad \epsilon \in N(0, \sigma^2)$$ Variance now is an surface in 3D, not just

More than 1 independent variable:
Normality? Are points linear? Homoscedastic?





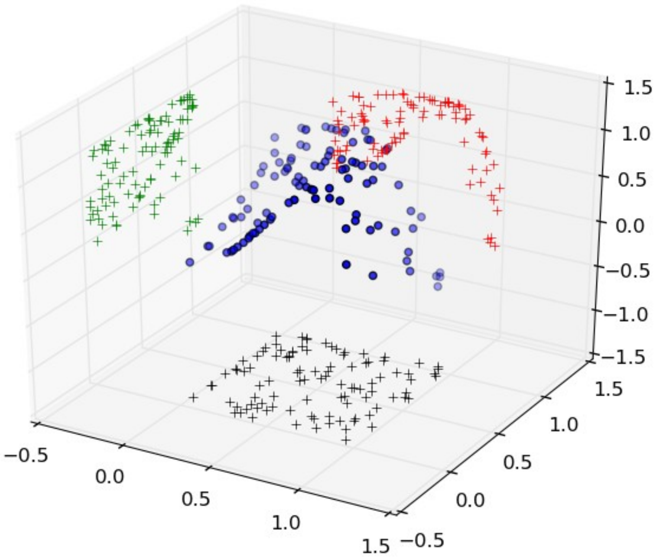


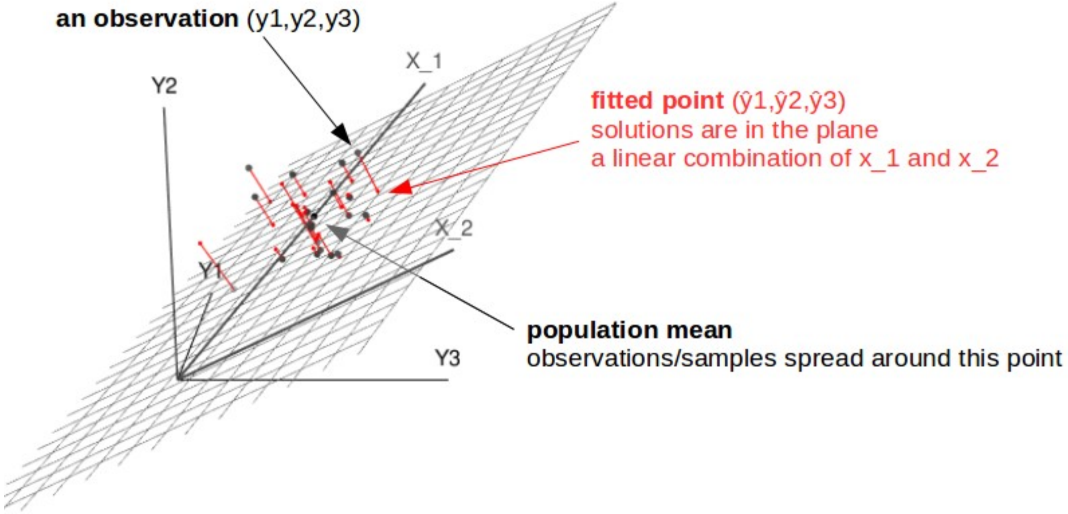
Key Idea: project all variables into 2D and check all variables at once

> Normality/Homosced does not depend on surface shape, only
    on deviations from this surface in Y axis = residuals
    * You can see them as lines on the left image

> If I were to replace all X variable axes with single residual axis,
    I do not lose information about these 2 properties

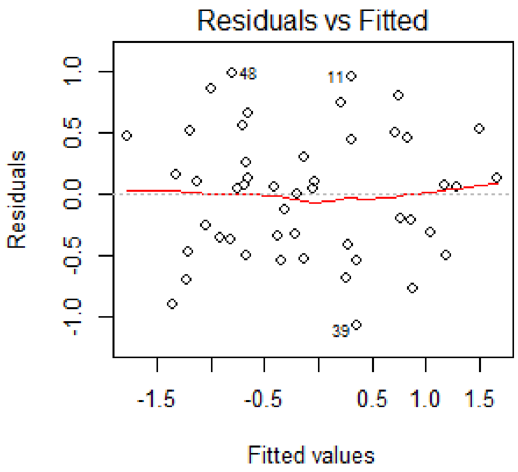
Why? Intuition:
> Normality/Homoscedasticity in 3D (or higher) = Normality/Homoscedasticity in EVERY AXIS X



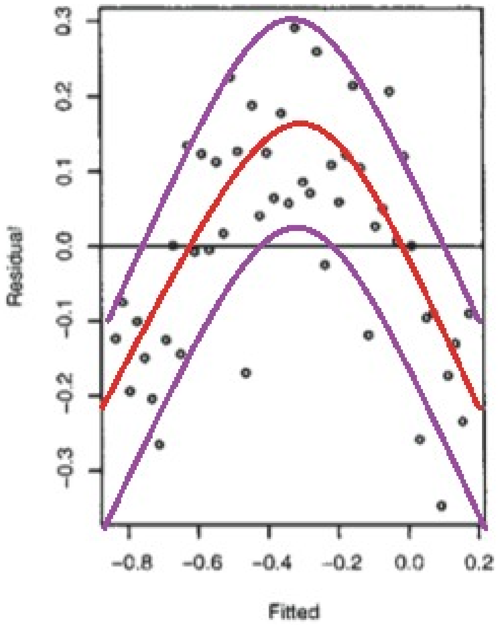


First plot: Residual plot
Check for linearity

~ almost linear

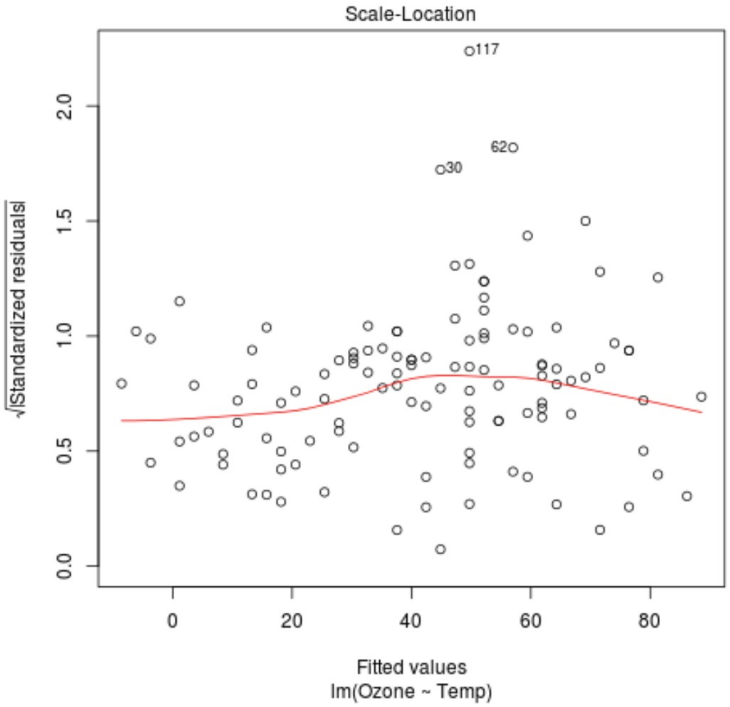


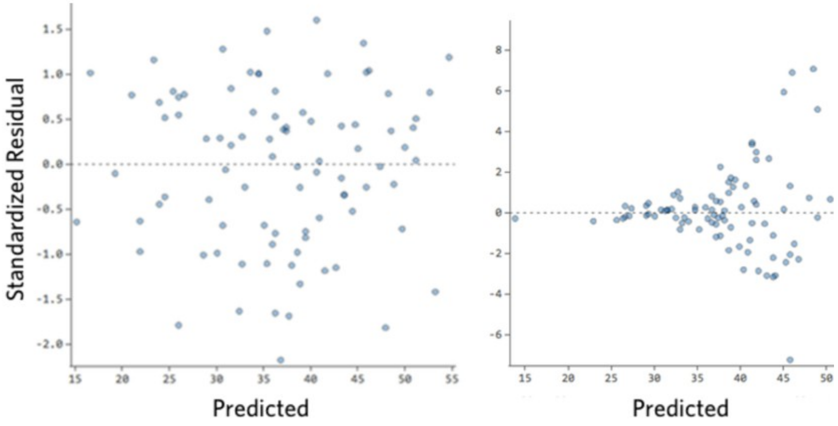
~ non-linear



If non-linear ---- use polynomial regression (later)

Third plot: Residual plot
Check for homoscedasticity (same variance)



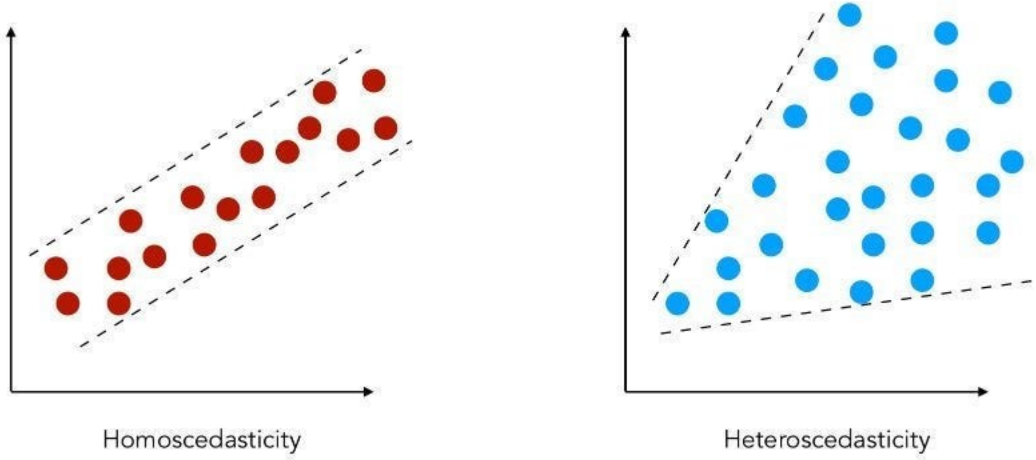


Horizontal line = homoscedastic (same variances)

Any non-horizontality = heteroscedasticty (difference in variance)

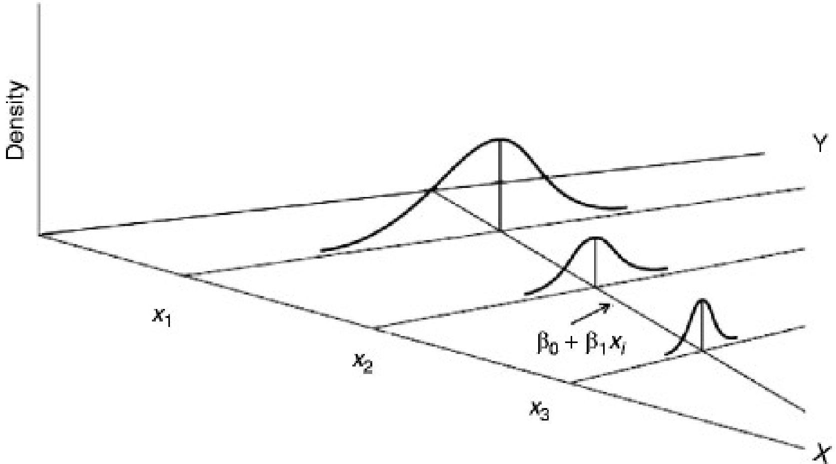
Alternatively just use Breusch-Pagan test

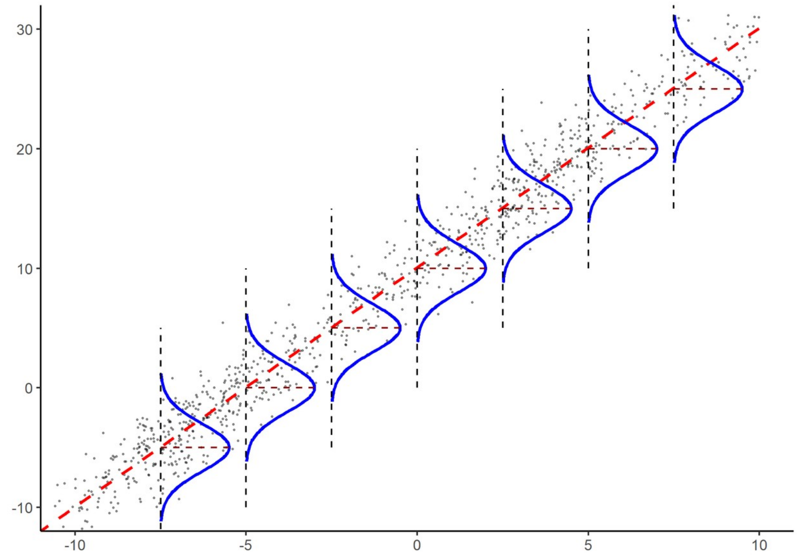
If heteroscedastic (different variances) ---- use weighted linear regression or some transformation of data (log, sqrt, ...)



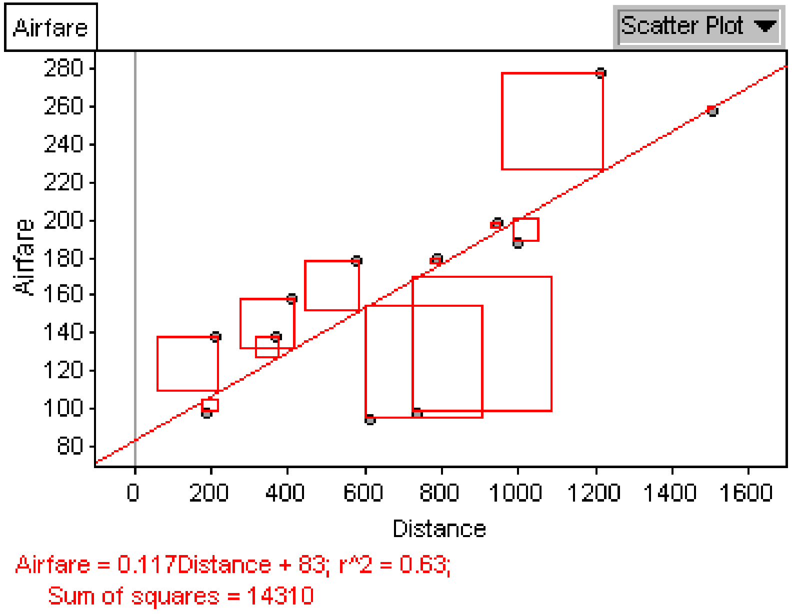
Homoscedasticity (same variance per X)

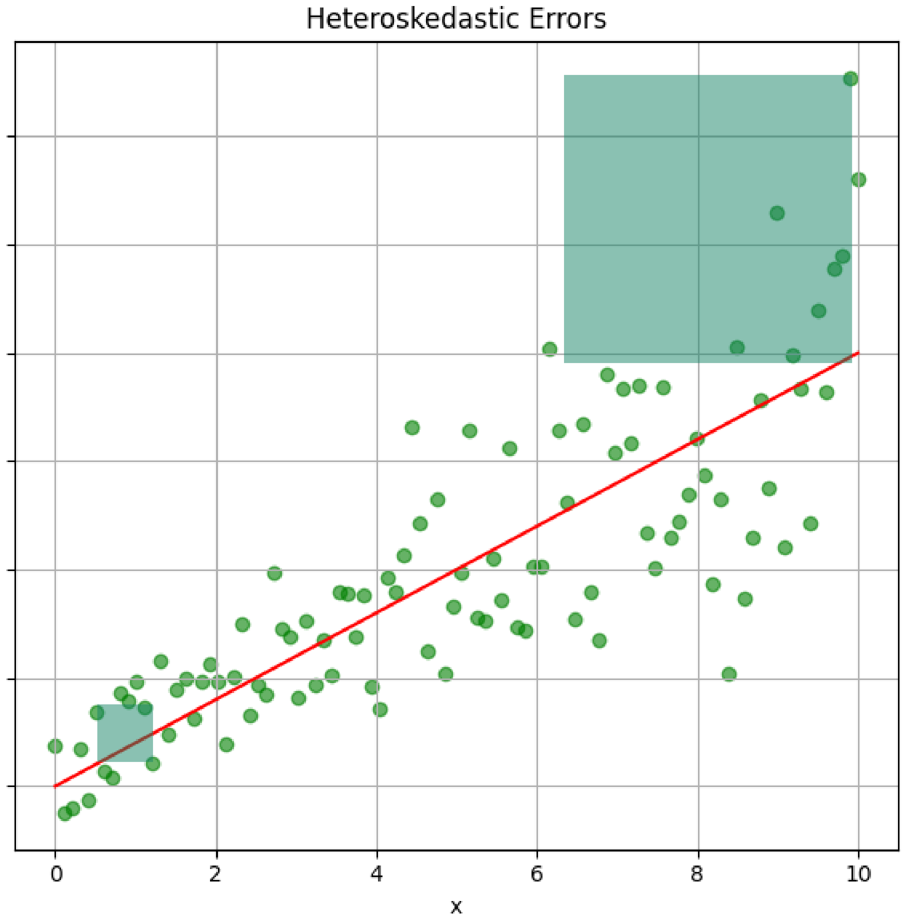
Heteroscedasticity (different variance per X)

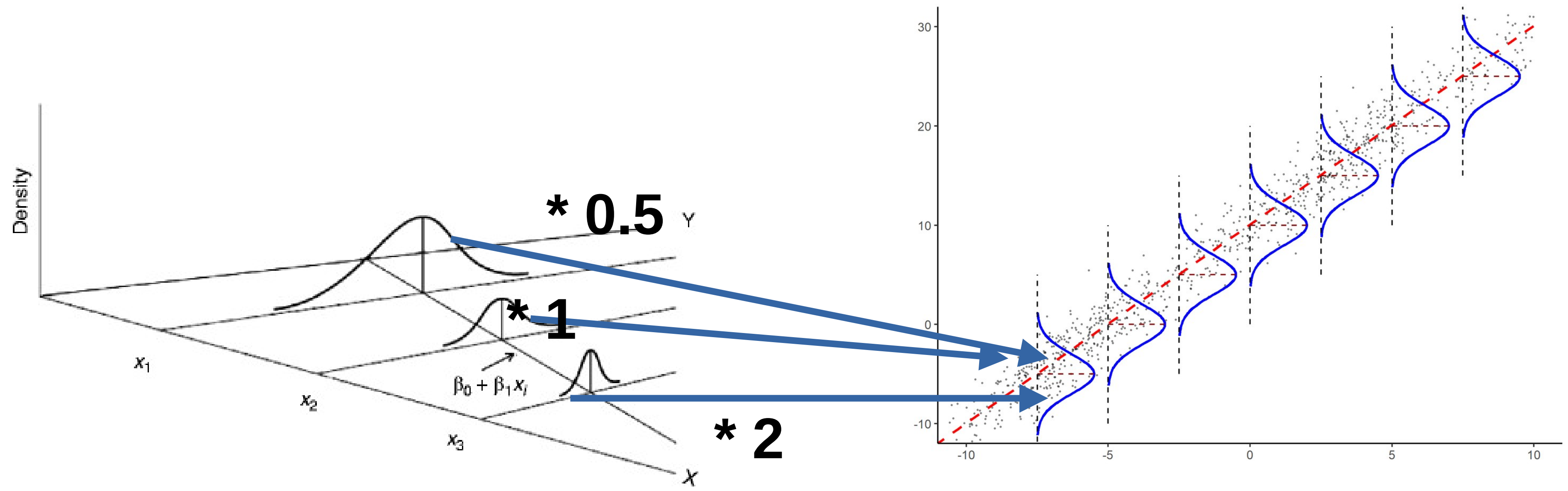




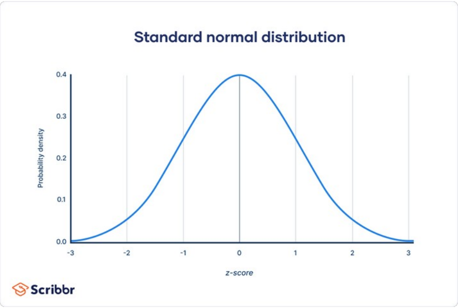
Problem with heteroscedasticity:
larger variances have more effect on line direction



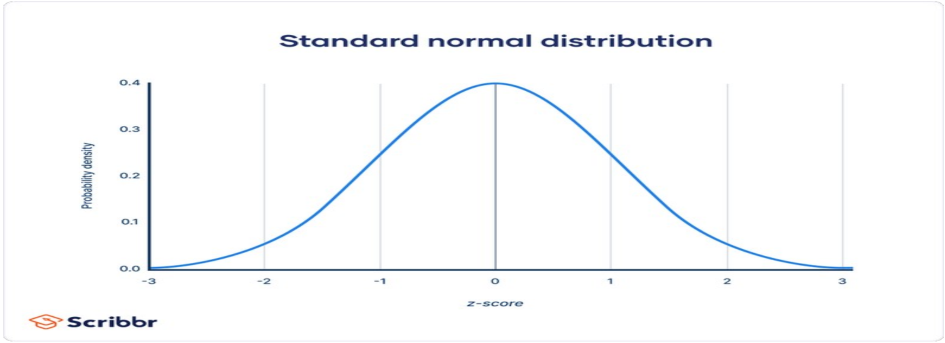




2 *

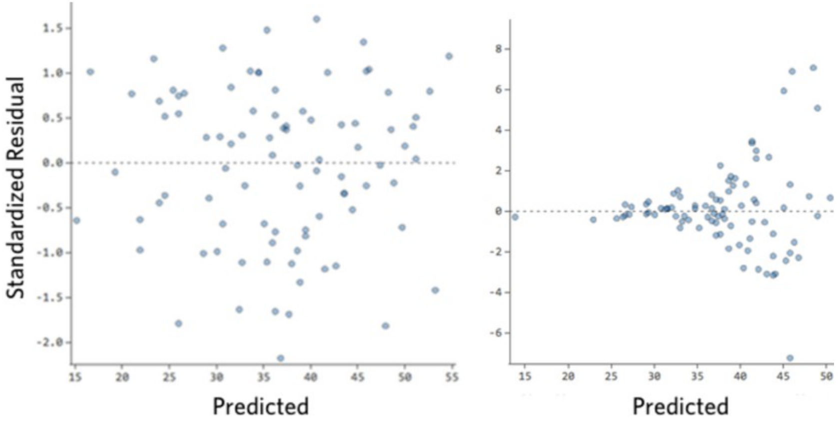


=



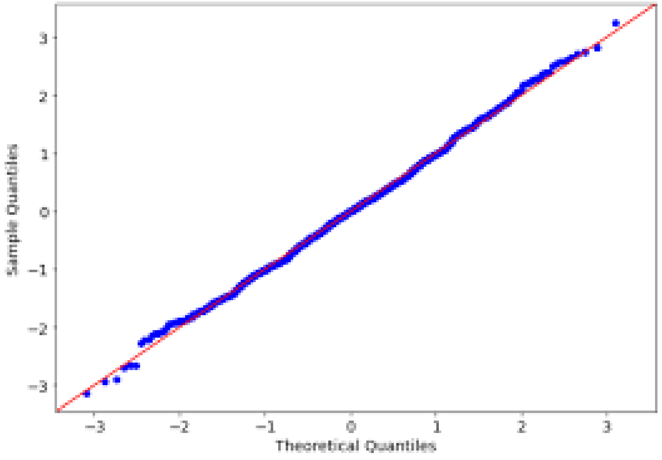
Idea of weighted linear regression / Weighted Least Squares (WLS)

Second plot: Q-Q plot
Check for normality

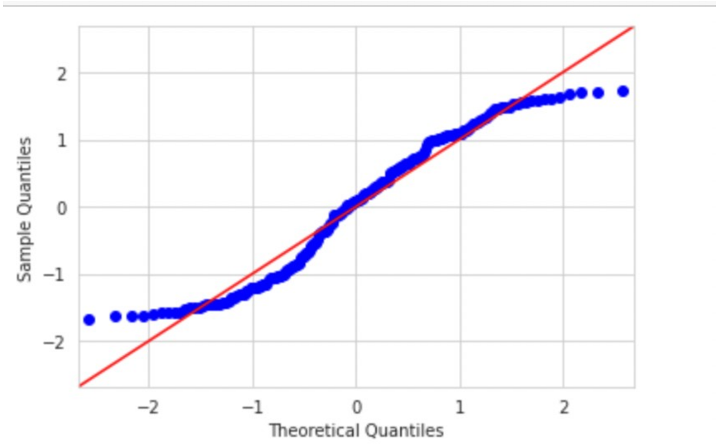


Call QQPlot()

Surely is normally distributed



Not quite normal



How to use it in R:

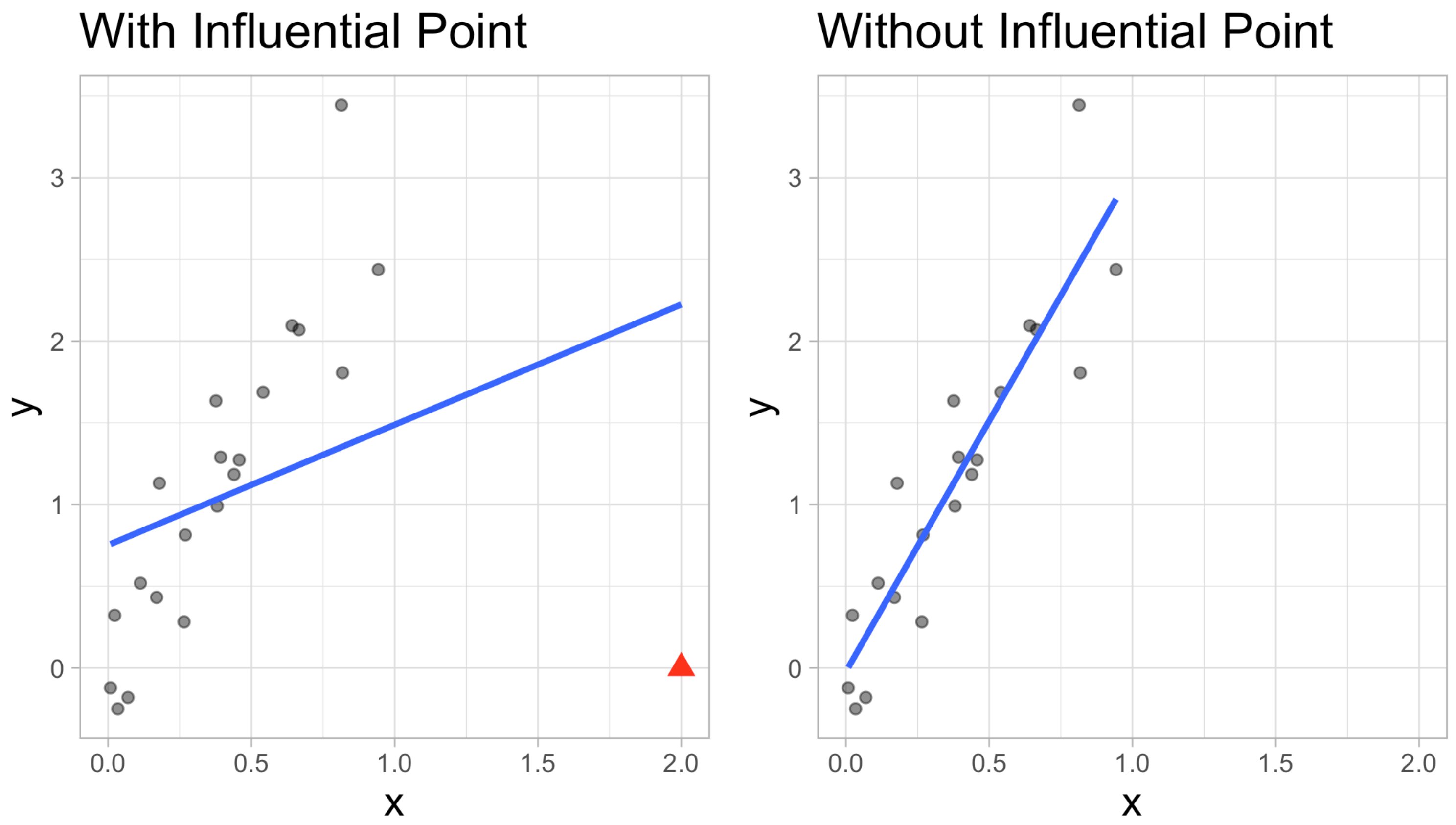
$$\min \sum_{i=1}^{n} w_i \cdot (y_i - \hat{y}_i)^2$$

```
# First: "Vanilla" OLS estimation
fit_OLS <- lm(y ~ ., data = df)

# Second: Weights
weights <- 1 / fitted(lm(abs(residuals(fit_OLS)) ~ fitted(fit_OLS))) ^ 2

# Third: Weighted least squares estimation
fit_WLS <- lm(y ~  ., data = df, weights = weights)
```

https://stackoverflow.com/questions/74417257/how-do-i-set-the-weighed-in-linear-regression-model-in-r

Fourth plot: Influential point plot
Check for outliers/influential points

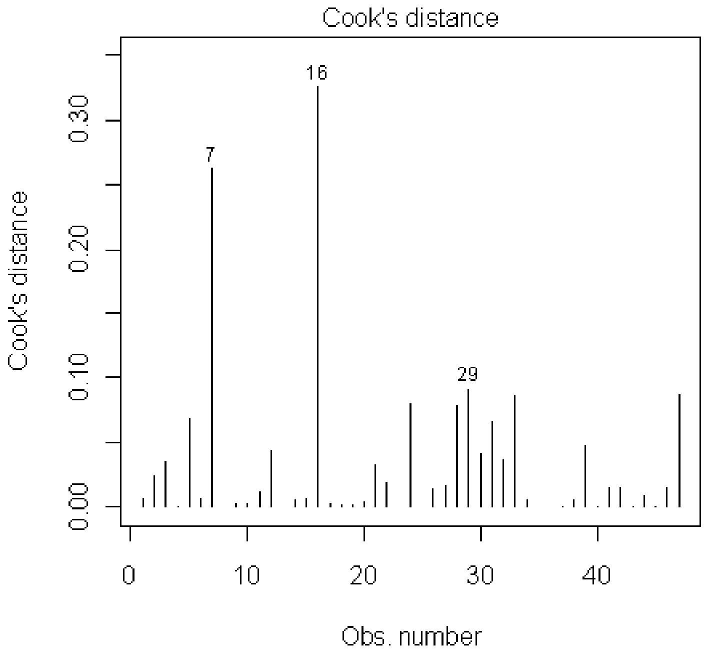
For every point: compare slope **with point** and **without point**.
If it is larger than a given threshold, we mark this point as **influential point**



How to use it in R:

```
full_lm = lm(y ~ ., data=df)
plot(full_lm, which = 4)
```

```
which(cooks.distance(full_lm) > 4 / (nrow(d) - length(coef(full_lm))))
```



Output:
7 16 29 ….

https://stats.stackexchange.com/questions/164099/removing-outliers-based-on-cooks-distance-in-r-language

Additional testing: multicollinearity

- we have already seen that correlations among predictors cause problems
  - coefficients fluctuate, interpretations become hazardous, overfitting,
- **multicollinearity** can be detected with variance inflation factor (VIF)
  - it measures the relationship between an independent variable and the other independent variables,

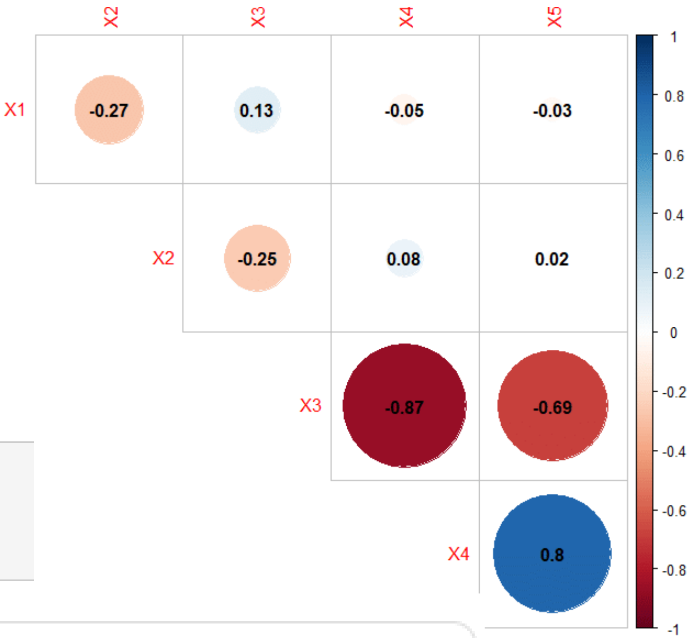
$$VIF_i = \frac{1}{1 - R_i^2}$$

    where R_i^2 is the coefficient of determination when the i-th independent variable is regressed on all the other independent variables in the model,
  - the predictors with large VIF likely do not improve the model, and could be removed.

How to use it in R:

```
cor_matrix <- cor(mtcars[c("disp", "hp", "wt", "drat")])
```

- Find out which pairs have linear connection between them



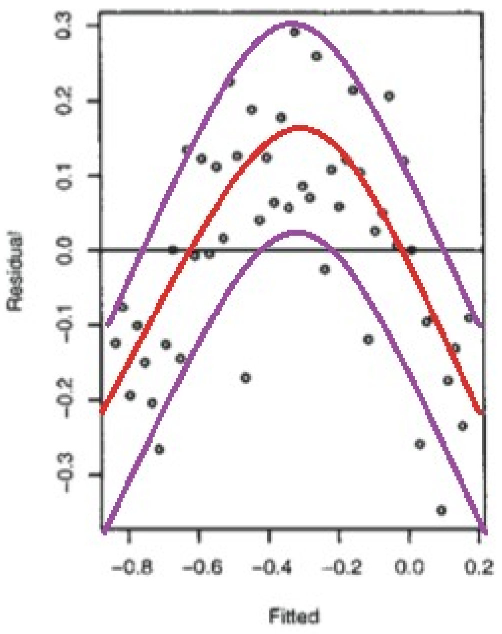

```
# detect potential multicollinearity
vif(full_lm)
```

```
    disp       hp       wt     drat
8.209402 2.894373 5.096601 2.279547
```

If value is **> 2.0** or **> 2.5**, strong multicollinearity, should remove the highest variable and repeat until no multicollinearity left

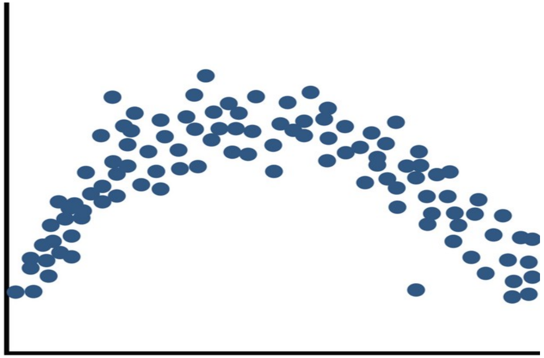
~ non-linear

First plot: Residual plot
Check for linearity



How to fit a non-linear model?
Use polynomial regression!

If I see something like:



Then I want a Polynomial of degree 2

From this:

$$Y=\beta_0+\beta_1\cdot X+\epsilon,$$
$$\epsilon\in N(0,\sigma^2)$$

It will look like this:

$$Y=\beta_0+\beta_1\cdot X+\beta_2\cdot X^2+\epsilon,$$
$$\epsilon\in N(0,\sigma^2)$$

How do I fit it? Use multivariate approach:

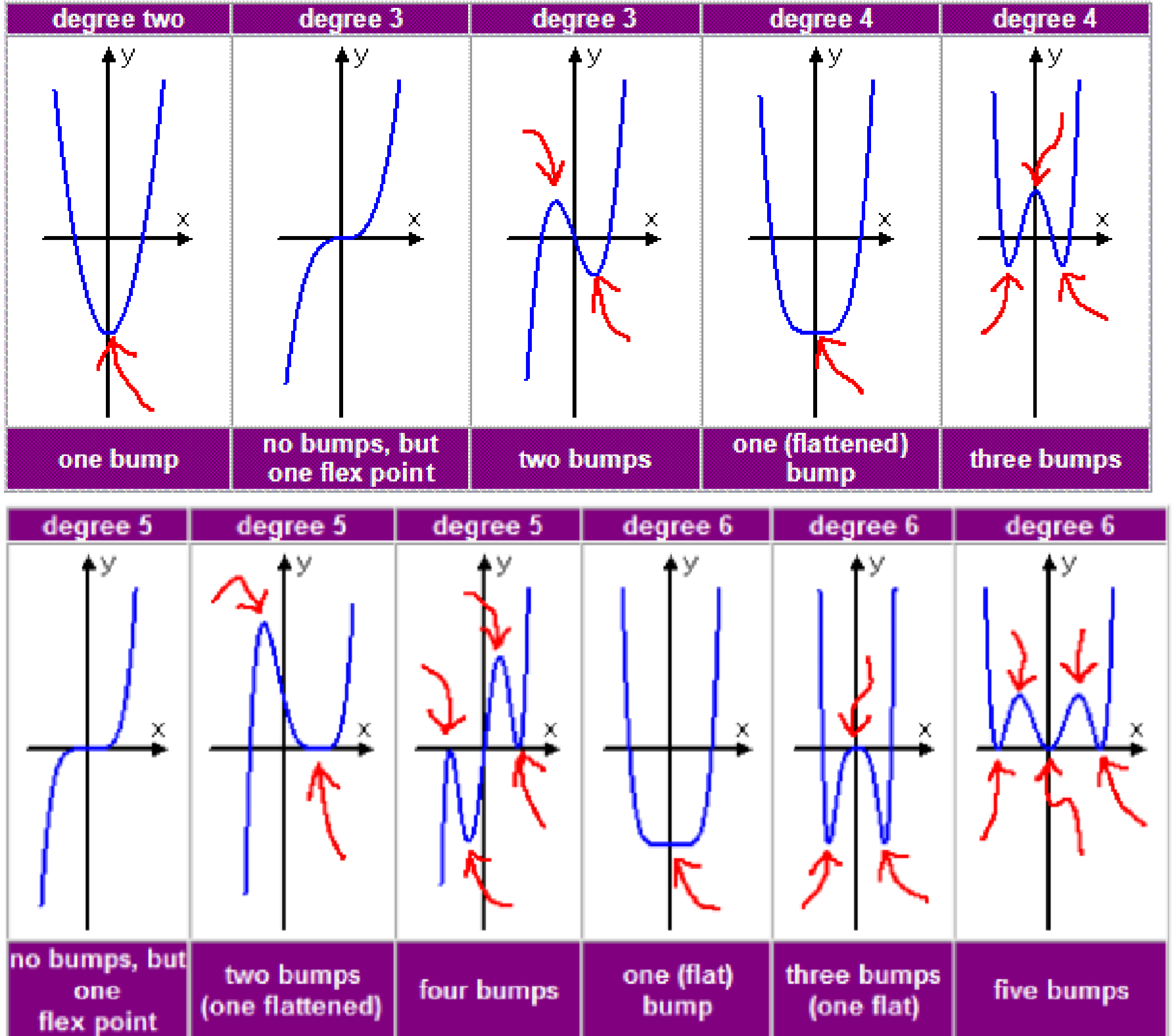
> ***creating synthetic data from given data and just run OLS algorithm again***

$$X_1=X$$
$$X_2=X^2$$
$$\ldots$$
$$X_P=X^P$$

$$Y=\beta_0+\beta_1\cdot X_1+\beta_2\cdot X_2+\ldots+\beta_P\cdot X_P+\epsilon,$$
$$\epsilon\in N(0,\sigma^2)$$

Polynomial regression

How to determine the degree of polynomial?



https://bhs229.weebly.com/uploads/5/7/9/8/5798388/notes_-_analyzing_and_sketching_polynomial_functions_smart_pdf.pdf

Some strange R poly() stuff? Well, not really:

$$X_1 = X \quad X_2 = X^2 \quad \dots \quad X_P = X^P$$

$$\vec{x} = \begin{pmatrix} 1 \\ 2 \\ \dots \\ 100 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 1^2 & 1^3 \\ 2 & 2^2 & 2^3 \\ .. & \dots & \dots \\ 100 & 100^2 & 100^3 \end{pmatrix}$$

But they are correlated!
May cause some problem,since:

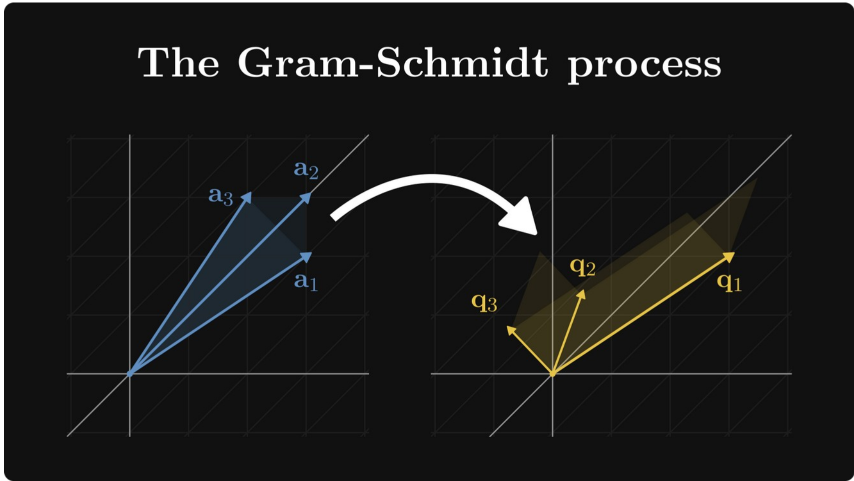
correlation = broken assumption of LM

$$\vec{x} = \begin{pmatrix} 1 \\ 2 \\ \dots \\ 100 \end{pmatrix} \rightarrow \begin{pmatrix} 1 & 1^2 & 1^3 \\ 2 & 2^2 & 2^3 \\ .. & \dots & \dots \\ 100 & 100^2 & 100^3 \end{pmatrix} \rightarrow \text{Gramm-Schmidt ortog} \rightarrow \begin{pmatrix} -0.17 & 0.22 & -0.25 \\ .. & \dots & \dots \\ .. & \dots & \dots \\ 0.15 & 0.14 & 0.21 \end{pmatrix} = \vec{X}_{ortog}$$

Even though they are changed = changed coefficients,

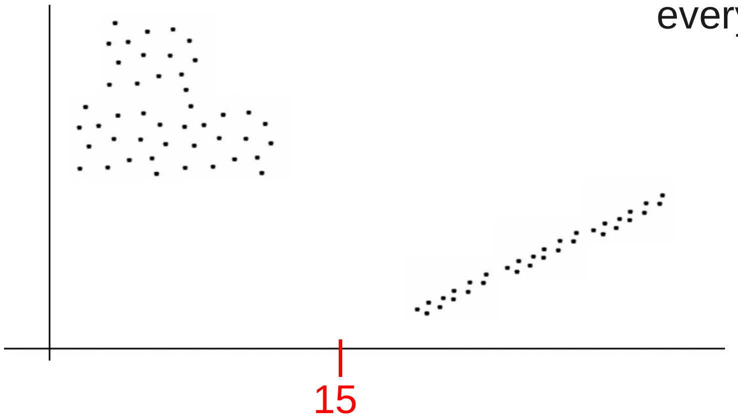
but their predictions are the same!

predict(lm with x coefs) = predict(lm with x_ortog coefs)



If I see something like:

Then I want two separate linear models on every interval



(knot) = interval change point

From this:

$$Y = \beta_0 + \beta_1 \cdot X + \epsilon ,$$
$$\epsilon \in N\left(0 , \sigma^2\right)$$

It will look like this:

$$Y = \beta_0 + \beta_1 \cdot X + \epsilon , if \ X < 15$$
$$Y = \beta_2 + \beta_3 \cdot X + \epsilon , if \ X \geq 15$$
$$\epsilon \in N\left(0 , \sigma^2\right)$$

How do I fit it? Train each interval separately, e.g., divide data into 2 separate LM calls

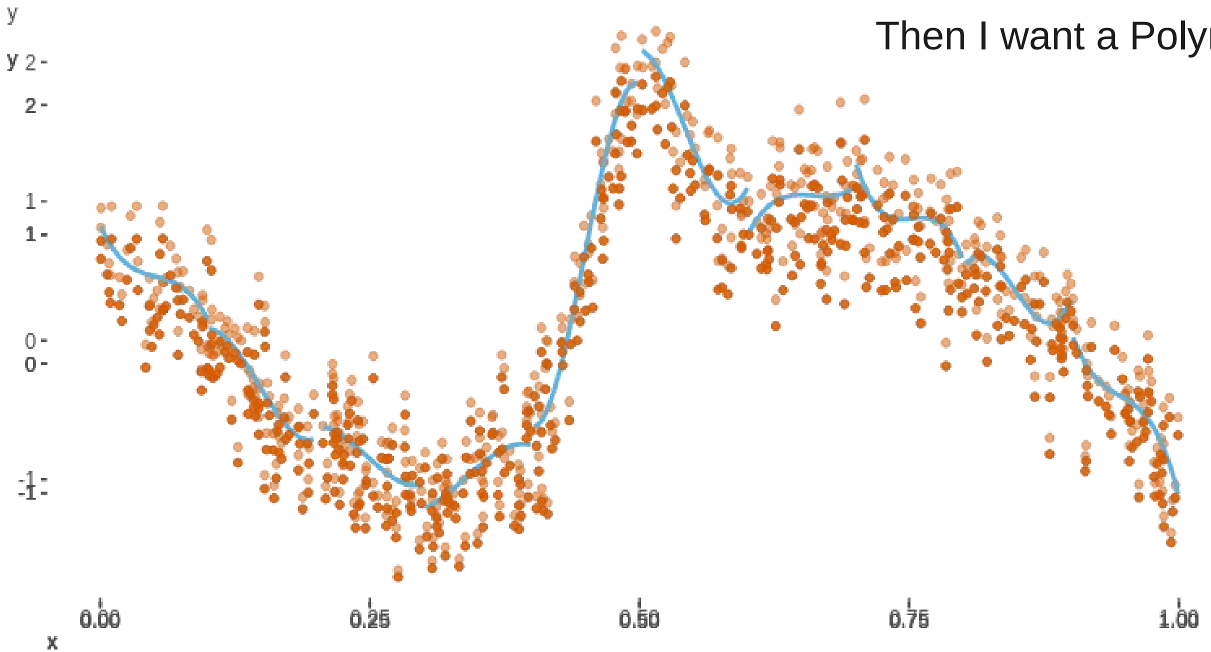
OR

Add a categorial switch variable for every interval change = knot

$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot \left(X \geq 15\right) + \epsilon$$
$$\epsilon \in N\left(0 , \sigma^2\right)$$

Step functions

If I see something like:



Then I want a Polynomial of large degree

$$X_1 = X$$
$$X_2 = X^2$$
$$\ldots$$
$$X_P = X^P$$

$$Y = \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + \ldots + \beta_P \cdot X_P + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

Piece-wise polynomial

But the shape is too complex, do I really need to do the P=12 degree polynomial???

Nope, combine previous two approaches: 1) Divide interval into several parts

2) Fit simple, say 3-degree polynomial in it

But what if I need to have a continuous function, not some unrelated polynomial parts?

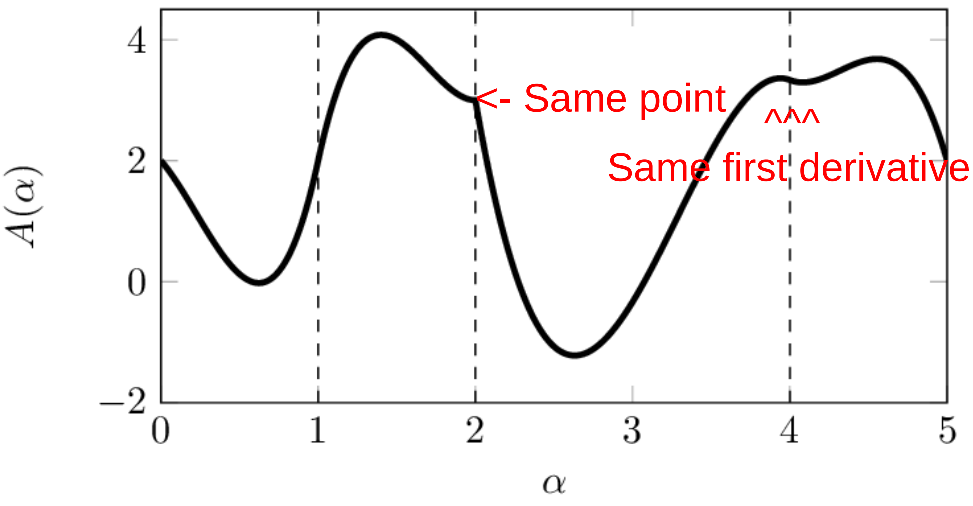
E.g.:

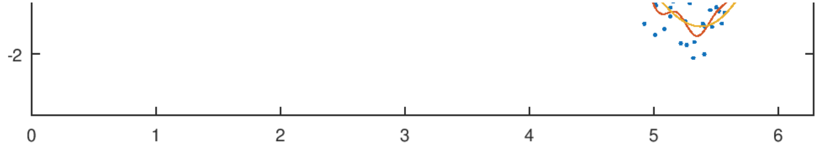
$$Y_1 = \beta_{10} + \beta_{11} \cdot X + \beta_{12} \cdot X^2 + \ldots + \beta_{1P} \cdot X^P + \epsilon, \quad \text{if } X < C$$

$$\epsilon \in N(0, \sigma^2)$$

$$Y_2 = \beta_{20} + \beta_{21} \cdot X + \beta_{22} \cdot X^2 + \ldots + \beta_{2P} \cdot X^P + \epsilon, \quad \text{if } X \geq C$$

$$\epsilon \in N(0, \sigma^2)$$





+ Continuity constraints:

$$Y_1(C) = Y_2(C) \ (\text{same point}, 0-\text{continuity})$$

$$Y_1'(C) = Y_2'(C) \ (\text{same first derivative}, 1-\text{continuity})$$

$$Y_1''(C) = Y_2''(C) \ (\text{same second derivative}, 2-\text{continuity})$$

TOO HARD!

Splines will ensure maximum continuity (d-1 for polynom with degree d) with an easy formula:

Splines

- we can represent this model with truncated power basis functions

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \cdots + \beta_{K+3} b_{K+3}(x_i) + \epsilon_i$$

- where the b_k are **basis functions**

$$b_1(x_i) = x_i$$ <

$$b_2(x_i) = x_i^2$$ <   construct first interval polynom

$$b_3(x_i) = x_i^3$$ <

$$b_{k+3}(x_i) = (x_i - \xi_k)_+^3 \quad k = 1, \ldots, K$$

< Every time we hit another knot,
We change all coeficients (beta0 – beta 3).
Largest degree (3 for cubic) encapsulates every other term inside:
x^3 - A x^2 + B x - C

- where

$$(x_i - \xi_k)_+^3 = \begin{cases} (x_i - \xi_k)^3 & \text{if } x_i > \xi_k \\ 0 & \text{otherwise} \end{cases}$$

We don’t have time to cover the math, refer to the lectures!

We don’t have time to cover the smoothing splines either!

Okay, I have a [Polynomial/Step function/Piece-wise poly/Spline/…….],

but how do I know which

[degree of polynom/ number of interval points(knots) / …/ any other non-LinearReg parameter]

To choose?

Cross validation!

Choose a set of models: $degree=4, degree=5, \ldots, degree=12$

$number\ of\ intervals: 4, 5, 6, \ldots, 11$

1) Split data into 10 folds

$model_1 = [degree=4, knots=4],$

$model_2 = [degree=4, knots=5], \ldots.$

2) For every model:

For every possible fold id(1, 2, …, 10):

2.0) Combine 9/10 folds intro train, 1/10 into test set

2.1) fit on 9/10 train folds

2.2) Compute RSS/F-stat/adjusted R^2 on 1/10 test fold

3) Average test errors from 2.2) for across all fold id

4) Choose the best model by averaged score from 3)

Or you could proceed with ANOVA/F-test comparison or other way of comparing **nested** models

$$Y = \beta_0 + \beta_1 \cdot X + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

$$Y = \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

=====

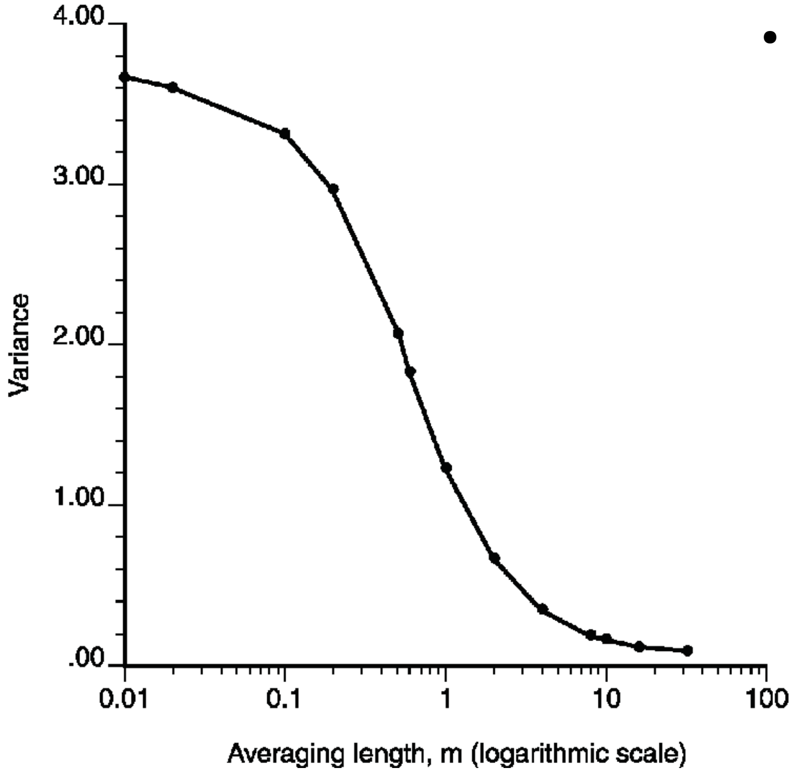
$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot X^2 + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

$$Y = \beta_0 + \beta_1 \cdot X_1 + \beta_2 \cdot X_2 + \beta_3 \cdot X_3 + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

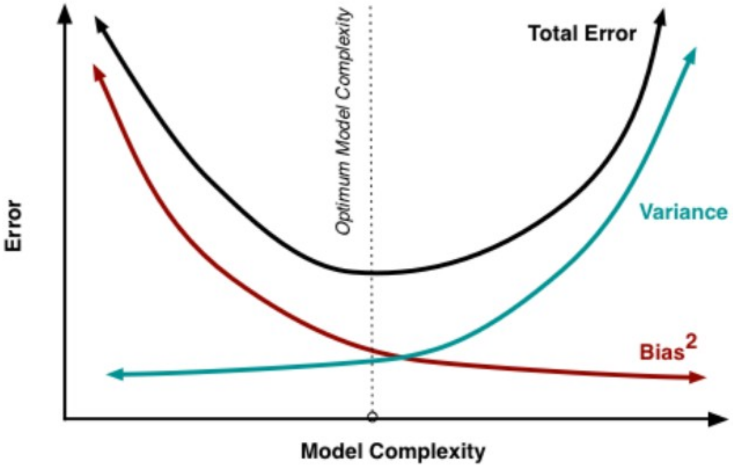
=====

$$Y = \beta_0 + \beta_1 \cdot X + \beta_2 \cdot X^2 + \beta_3 \cdot X^3 + \epsilon,$$
$$\epsilon \in N(0, \sigma^2)$$

And then just use **AIC** / **F-stat as in ANOVA** / **adjusted R^2**



- You can think of this as following:
  - Addition of more variables (**or degrees in polynomial**) makes the prediction more precise (smaller variance)
  - But decrease from 1.1% to 1.09% is too small and the computational complexity is spent for nothing
  - Moreover, with more variables we need more and more samples to maintain same variance quality
    - Bias-variance tradeoff
    - **Is exactly how Cross-validation works!**

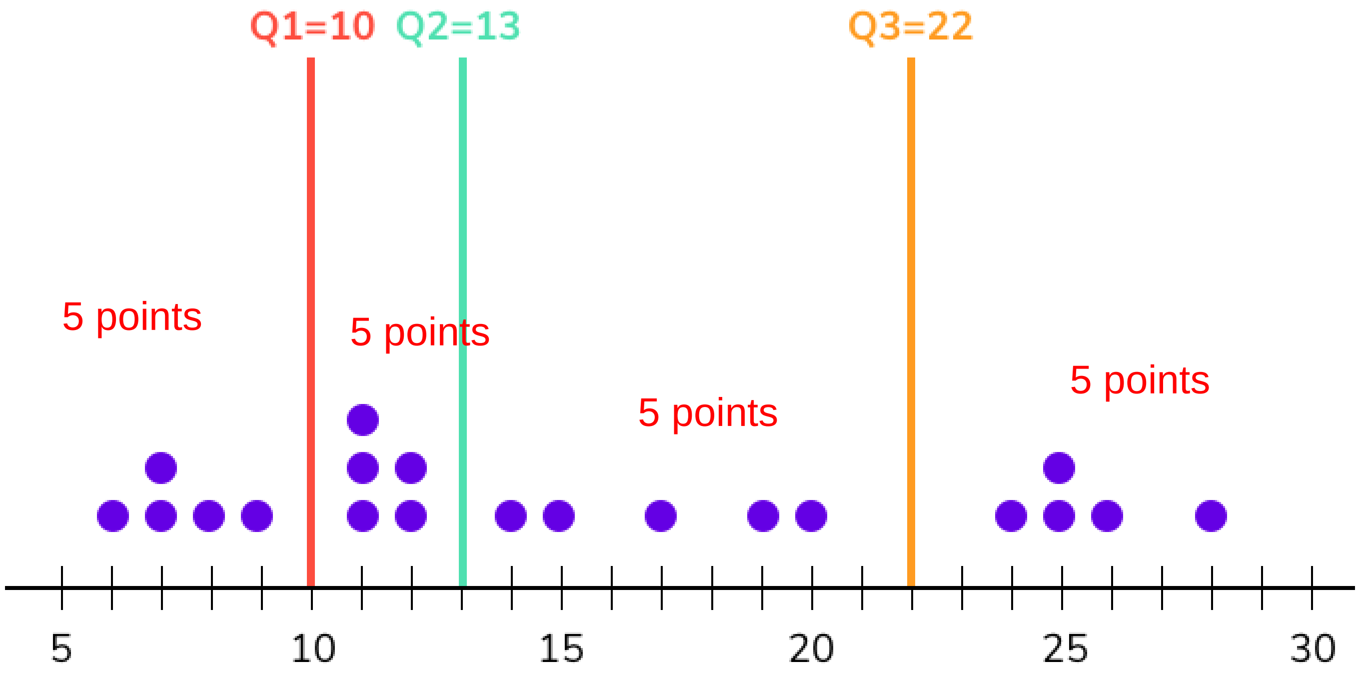


We can choose the position of knots if number of them is given:

quantiles =

such that each interval between knots has the same number of points

> optimality control via Cross-Validation



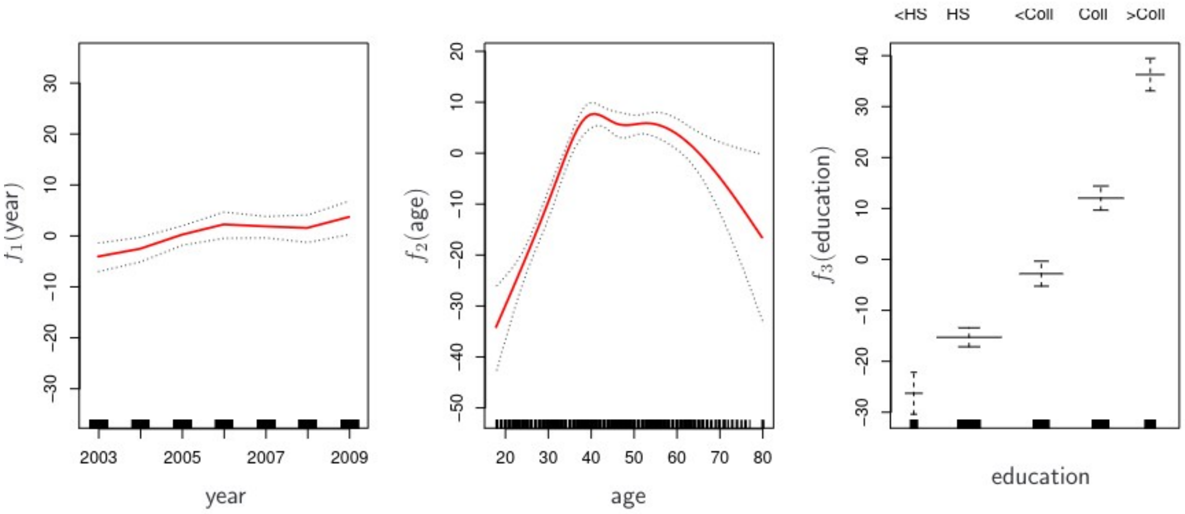
Okay, finally I can do it for 1 variable, but what if two variables have different models? How to combine them?

> Just add them and assume additivity(no interactions)

$$y_i = \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}) + \epsilon_i$$

> Or, if interaction are REALLY needed,
R library will do it for you!

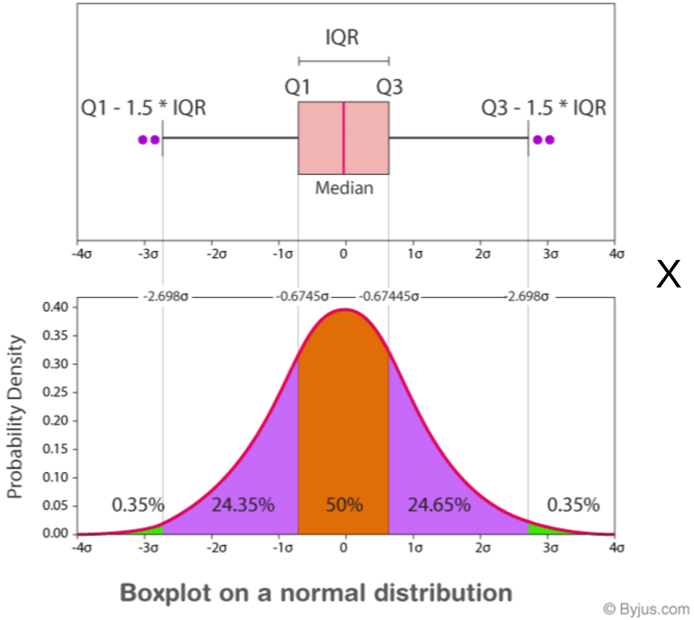
gam(… + ns(age,df=5):ns(year,df=5))



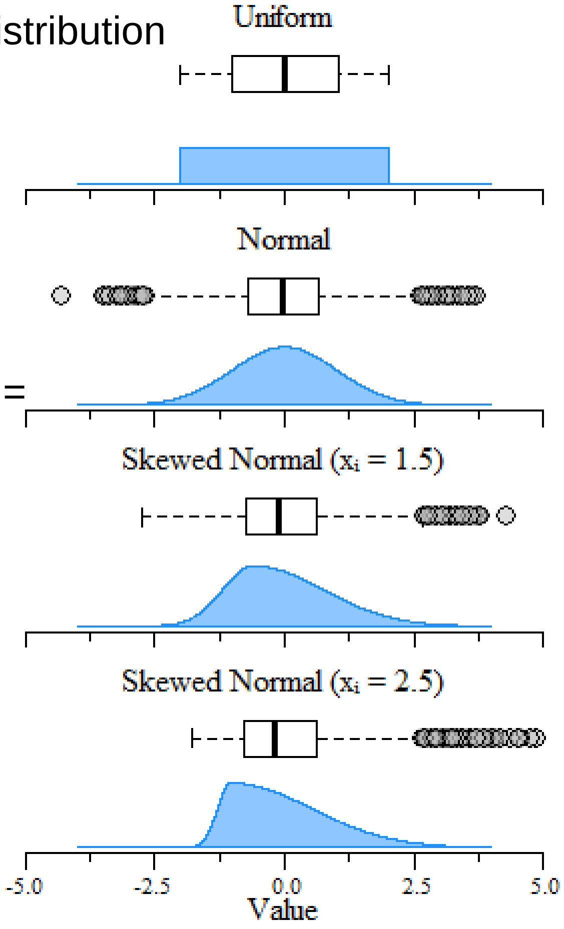
gam(
wage ~ ns(year, df = 5) + ns(age, df = 5) + education
)

Generalized Additive Models (GAMs)

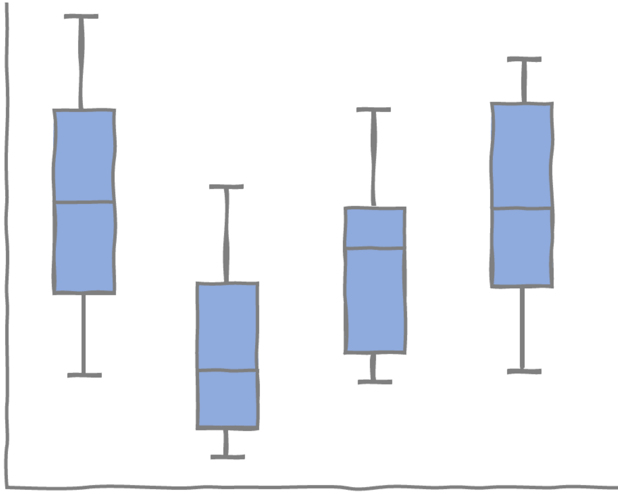
Some strange plot? **Boxplot** = categorical X distribution

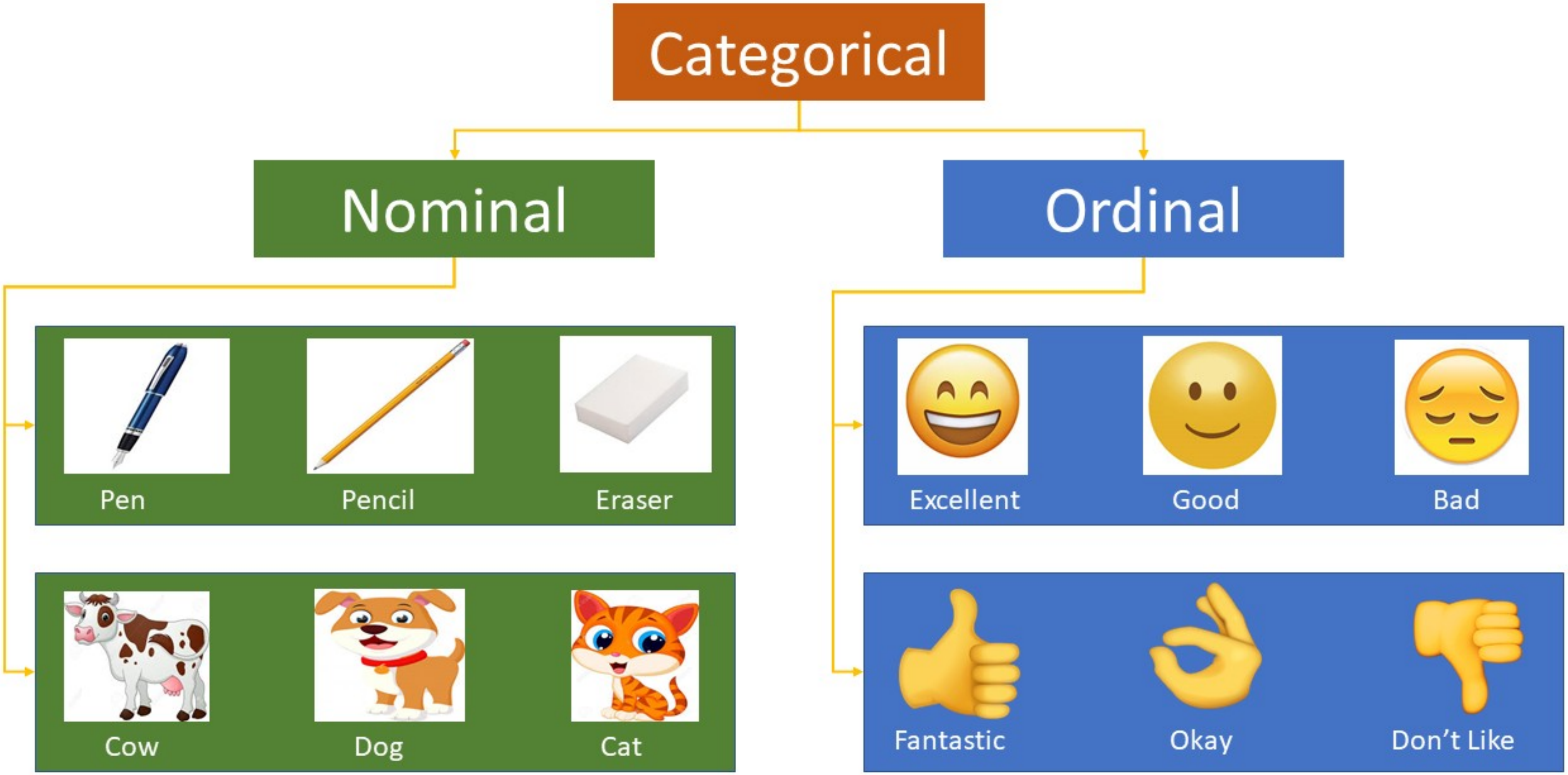


X 3 groups =



Remove distribution plots

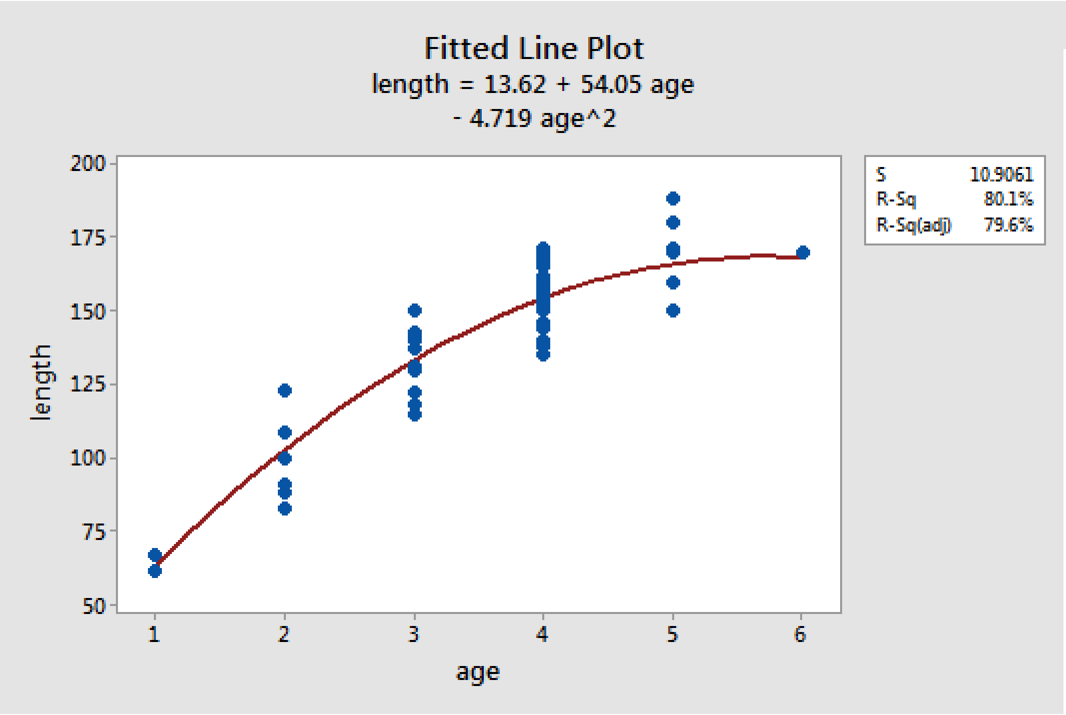


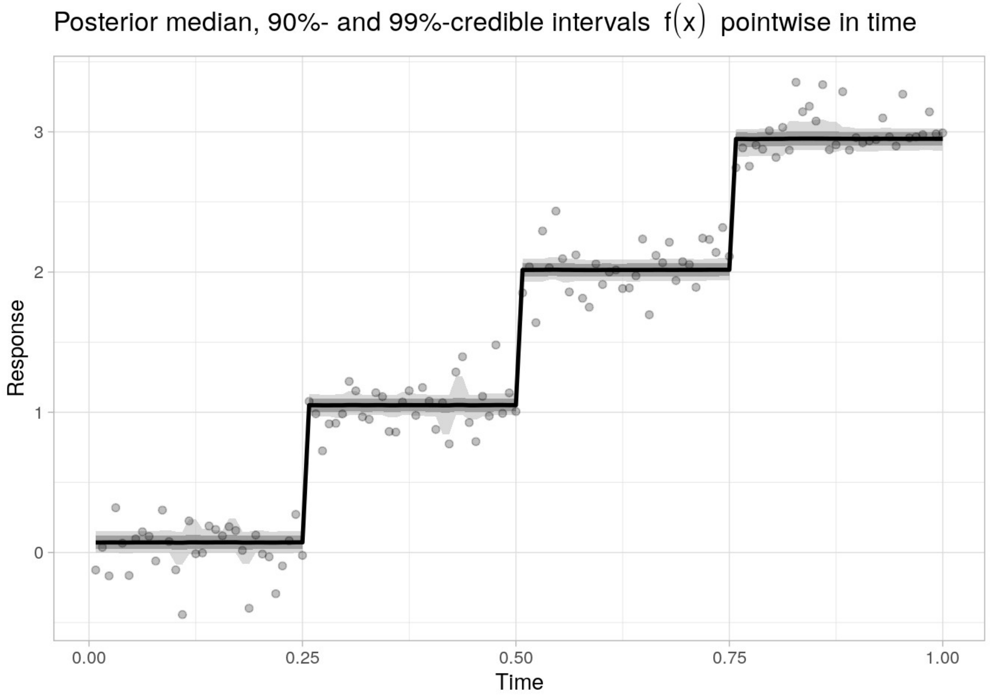



```
df$animal <- as.factor(df$animal)
```

```
df$reaction <- as.factor(df$reaction)
df$reaction <- ordered(df$reaction,
levels = c("fantastic", "okay",
"dontlike"))
```

Categorical(Discrete) X? Just use Polynomial or Step function)





Please open the today's activity .zip file and find the:

4	13.10.	JB, AA, JK	Non-linear regression	san_nonlinear.zip, pres_4.pdf

You will find the activity (voluntary) questions at the end:

Further questions to answer (homework):

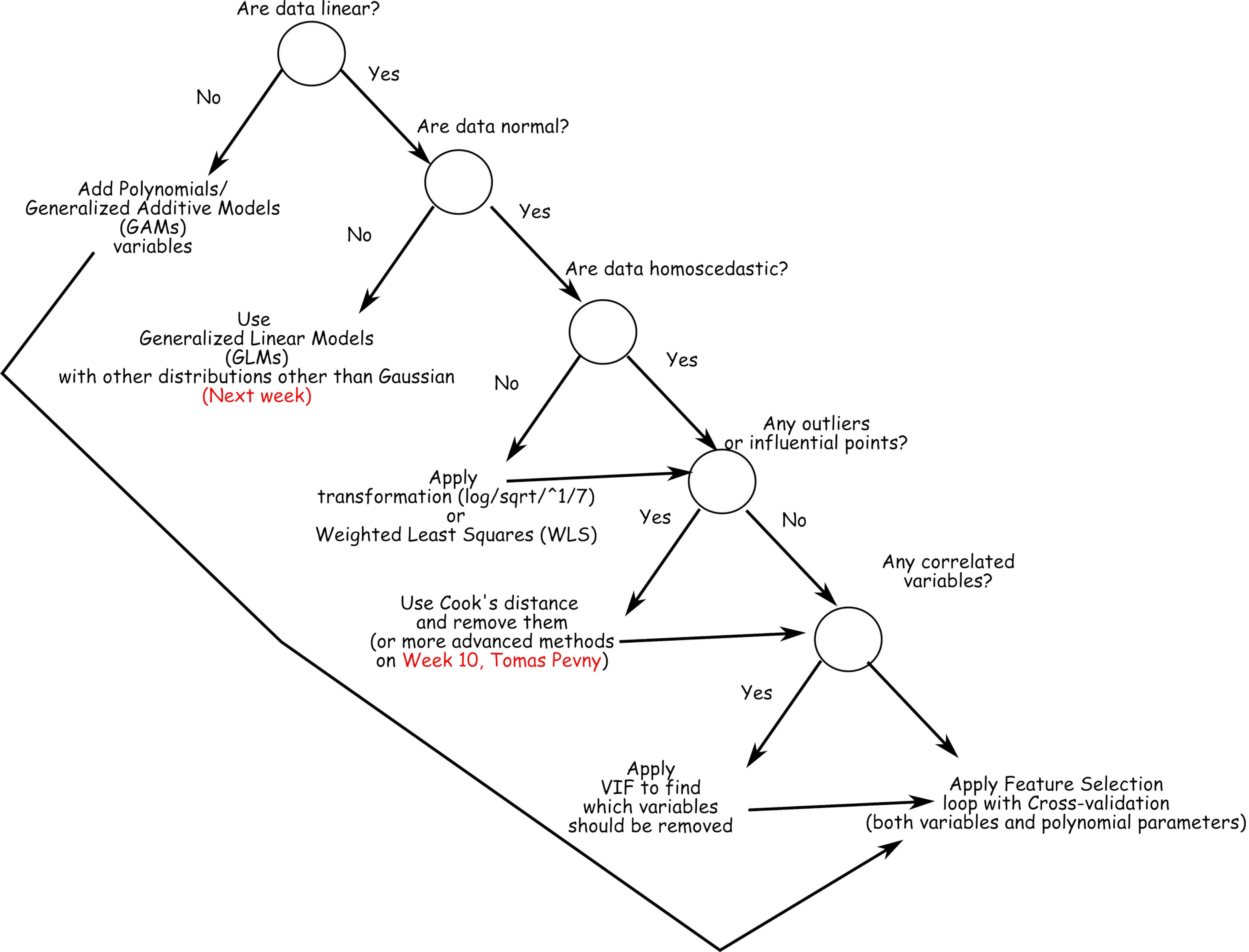
1. Have a look at other predictors. What treatment would you recommend for them?
2. Which non-linear model would you recommend for wage prediction (considering all the predictors)? Show a model that improves the gam model tested in the last chunk.

Deadline: 20.10.2025 (next seminar, voluntary, 1 point)

- **Today a big homework will be presented now and the countdown starts**
  - **You will have ~ 2 weeks to complete it, the time is parallel to every-week small HW**

3	7.10.	JB, AA, JK	Shrinked linear regression	san_fs.zi, assignment1.zip	lab3.zip Linear regression.pptx

Deadline: 20.10.2025 (two weeks, full points)
27.10.2025 (three weeks, half points)



Are data linear?
No
Yes
Add Polynomials/
Generalized Additive Models
(GAMs)
variables
Are data normal?
No
Yes
Use
Generalized Linear Models
(GLMs)
with other distributions other than Gaussian
(Next week)
Are data homoscedastic?
No
Yes
Apply
transformation (log/sqrt/^1/7)
or
Weighted Least Squares (WLS)
Any outliers
or influential points?
Yes
No
Use Cook's distance
and remove them
(or more advanced methods
on Week 10, Tomas Pevny)
Any correlated
variables?
Yes
Apply
VIF to find
which variables
should be removed
Apply Feature Selection
loop with Cross-validation
(both variables and polynomial parameters)