Seminar 2

Martin Søyland

Disposisjon

1. Repetisjon
2. Korrelasjon
3. Bivariat OLS
4. Multipel OLS
5. Samspill og andregradsledd
6. Logistisk regresjon

Fokus: Regresjon og tolkning + litt visualisering!

Datasett

Sååå, samme datasett som sist. Husk at du må `setwd()` hver gang du åpner R (med mindre du bruker prosjekt). Data er passasjerer fra Titanic og variabler på om de overlevde, klasse, pris, osv. Dere kan enten laste ned data ved å skrive inn nettaddressen under i nettleseren og legge denne filen i mappen dere jobber fra:

```
setwd("~/Der/du/vil/jobbe/fra")

passengers <- read.csv("titanic.csv", stringsAsFactors = FALSE)
```

Jeg laster bare direkte inn fra linken. Legg merke til argumentet `stringsAsFactors = FALSE`. Dette står som default til `TRUE`. Argumentet konverterer alle variabler (kolonner) til klassen `factor()`, som er tilnærmet det samme som ordinalt målenivå – det vil vi ikke! Hvorfor vil vi ikke? Fordi vi vil ha lavest målenivå og heller sette det opp om vi finner ut at det gir mening, gitt data og det vi skal gjøre.

```
# passengers <- read.csv('https://folk.uio.no/martigso/stv4020/titanic.csv',
# stringsAsFactors = FALSE)

passengers <- read.csv("../data/titanic.csv", stringsAsFactors = FALSE)
class(passengers$Name)
```

```
## [1] "character"
```

Litt omkoding

Vi skal, som forrige gang, sentrere variabelen `Age`

```
median(passengers$Age, na.rm = TRUE)
```

```
## [1] 28
```

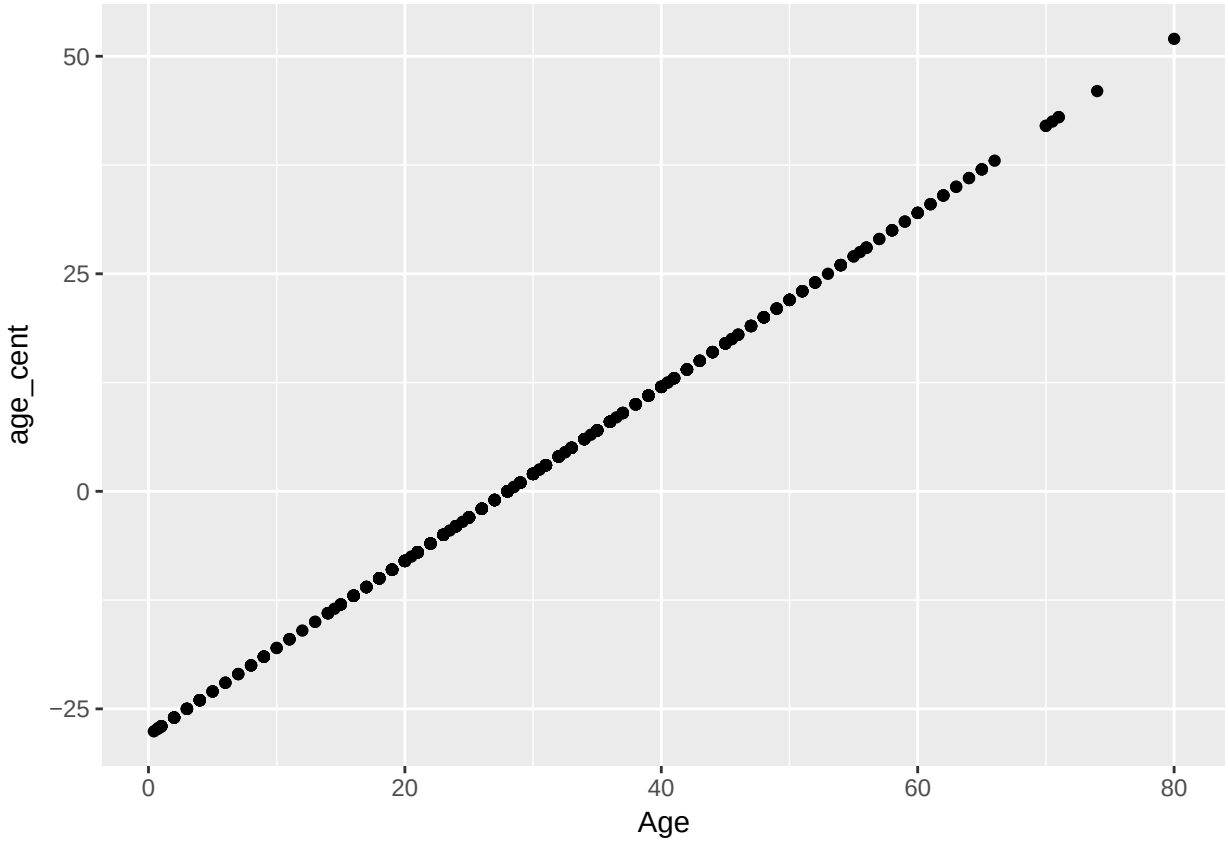
```
passengers$age_cent <- passengers$Age - median(passengers$Age, na.rm = TRUE)
```

Dette er en veldig god anledning til å se litt på **pakker**. R har nemlig et helt insane stort *open source* bibliotek med brukerlagede pakker alle har lov å bruke. Vi installerer en pakke med funksjonen `install.packages()` (husk å ha pakkenavnet i hermetegn her). Det er faktisk ikke nok å bare installer pakken, vi må også pakke den opp. Det gjør vi med `library()`. Pakken vil da være lastet *inn* til du avslutter R-sessionen du har åpen. Sååååå, "ggplot2" er en pakke for å lage grafikk, som vi kommer til å bruke mye (R har også en innebygd grafikk-fuksjon: `plot()`).

Nedenfor sjekker jeg om omkodingen vi gjorde er riktig. Syns dere det ser sånn ut?

```
# install.packages(ggplot2)
library(ggplot2)
ggplot(passengers, aes(x = Age, y = age_cent)) + geom_point()
```

```
## Warning: Removed 177 rows containing missing values (geom_point).
```



Vi kan også gjøre grafikken mye finere, men denne figuren vil ikke bli brukt i et evt paper. Så det er greit at den er litt quick and dirty. Kommer tilbake til det senere.

Korrelasjon

La oss også sjekke korrelasjonen mellom flere av variablene våre. Her bruker vi funksjonen `cor()` for bare korrelasjonsestimater.

Som dere husker fra forrige gang, må vi håndtere missingverdier. Men med korrelasjon er det, som dere vet, forskjellige måter å håndere missing på: pairwise og listwise exclusion. Dette er ikke viktig med korrelasjon mellom bare to variabler, men med flere variabler er det viktig:

```
cor(passengers[, c("age_cent", "Survived", "Fare")], use = "complete.obs")

cor(passengers[, c("age_cent", "Survived", "Fare")], use = "pairwise.complete.obs")
```

```
##             age_cent    Survived       Fare
## age_cent  1.00000000 -0.07692265 0.09638814
## Survived -0.07692265  1.00000000 0.27128592
## Fare      0.09638814  0.27128592 1.00000000
##             age_cent    Survived       Fare
## age_cent  1.00000000 -0.07722109 0.09638814
## Survived -0.07722109  1.00000000 0.25965960
## Fare      0.09638814  0.25965960 1.00000000
```

Multivariat OLS

Forrige gang kjørte vi en bivariat regresjon med `Survived` som avhengig variabel og `age_cent` som uavhengig variabel:

```
pass_reg <- lm(Survived ~ age_cent, data = passengers)
summary(pass_reg)
```

```
##
## Call:
## lm(formula = Survived ~ age_cent, data = passengers)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.4811 -0.4158 -0.3662  0.5789  0.7252
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.410601   0.018476  22.224   <2e-16 ***
## age_cent    -0.002613   0.001264  -2.067   0.0391 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4903 on 712 degrees of freedom
```

```
##   (177 observations deleted due to missingness)
## Multiple R-squared:  0.005963,   Adjusted R-squared:  0.004567
## F-statistic: 4.271 on 1 and 712 DF,  p-value: 0.03912
```

Vi ble likevel enige om at dette kanskje ikke var den beste spesifikasjonen av regresjonen, hvis vi undersøke hva som gjorde at passasjerene overlevde.

"Women and children":

```
pass_reg2 <- lm(Survived ~ age_cent + Sex, data = passengers)
summary(pass_reg2)
```

```
##
## Call:
## lm(formula = Survived ~ age_cent + Sex, data = passengers)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7786 -0.2115 -0.1931  0.2471  0.8401
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.7547117  0.0256497  29.424   <2e-16 ***
## age_cent    -0.0009206  0.0010730  -0.858    0.391
## Sexmale     -0.5469036  0.0323428 -16.910   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4144 on 711 degrees of freedom
##   (177 observations deleted due to missingness)
## Multiple R-squared:  0.2911, Adjusted R-squared:  0.2891
## F-statistic:   146 on 2 and 711 DF,  p-value: < 2.2e-16
```

Og noen personer er viktiger enn andre. . . :

```
pass_reg3 <- lm(Survived ~ age_cent + Sex + factor(Pclass), data = passengers)
summary(pass_reg3)
```

```
##
## Call:
## lm(formula = Survived ~ age_cent + Sex + factor(Pclass), data = passengers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.11410 -0.25081 -0.06422  0.23015  1.00676
##
## Coefficients:
```

```
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)       0.972139   0.033730  28.821  < 2e-16 ***
## age_cent         -0.005460   0.001084  -5.039 5.96e-07 ***
## Sexmale          -0.479456   0.030718 -15.608  < 2e-16 ***
## factor(Pclass)2 -0.207747   0.041689  -4.983 7.86e-07 ***
## factor(Pclass)3 -0.406618   0.038288 -10.620  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3849 on 709 degrees of freedom
##   (177 observations deleted due to missingness)
## Multiple R-squared:  0.3902, Adjusted R-squared:  0.3867
## F-statistic: 113.4 on 4 and 709 DF,  p-value: < 2.2e-16
```

Så kan man tenke seg at man prioriterte både de yngste og de elste i livbåter, fordi personer mellom har større sannsynlighet for å overleve uten livbåt (dog kanskje ikke så veldig stor absolutt sannsynlighet likevel). Dette kan vi teste med et andregradsledd. For å lage andregradsledd er det to alternativer, her er ett: (det andre er å bruke funksjonen `poly()`)

Andregradsledd (polynomer)

```
passengers$age_cent_andregrad <- passengers$age_cent^2

andregrads_reg <- lm(Survived ~ age_cent + age_cent_andregrad + Sex + factor(Pclass),
                     data = passengers)

# andregrads_reg <- lm(Survived ~ poly(age_cent, 2, raw = TRUE) + Sex + factor(Pclass),
#                      data = passengers[which(is.na(passengers$age_cent) == FALSE), ])

summary(andregrads_reg)
```

```
##
## Call:
## lm(formula = Survived ~ age_cent + age_cent_andregrad + Sex +
##     factor(Pclass), data = passengers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.15891 -0.24944 -0.05217  0.23243  1.00981
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          9.564e-01  3.553e-02  26.920  < 2e-16 ***
## age_cent            -6.034e-03  1.158e-03  -5.211 2.47e-07 ***
```

```
## age_cent_andregrad  6.745e-05  4.820e-05   1.399    0.162
## Sexmale            -4.798e-01  3.070e-02 -15.629  < 2e-16 ***
## factor(Pclass)2    -2.042e-01  4.174e-02  -4.891 1.24e-06 ***
## factor(Pclass)3    -4.034e-01  3.833e-02 -10.523  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3846 on 708 degrees of freedom
##   (177 observations deleted due to missingness)
## Multiple R-squared:  0.3919, Adjusted R-squared:  0.3876
## F-statistic: 91.24 on 5 and 708 DF,  p-value: < 2.2e-16
```

```
# plot(andregrads_reg)
# termplot(andregrads_reg, se = TRUE, terms = 1)
```

Logistisk regresjon

Logistisk regresjon er veldig likt i oppbygning. Det er i familien **general linearized models** (`glm()`). Det viktige her er argumentet `family = "binomial"`, som spesifiserer at vi snakker om en binær avhengig variabel – kan også skrive `binomial(link = "logit")`.

```
pass_logit <- glm(Survived ~ age_cent + Sex + factor(Pclass),
                  data = passengers, family = "binomial",
                  na.action = "na.exclude")

summary(pass_logit)
```

```
##
## Call:
## glm(formula = Survived ~ age_cent + Sex + factor(Pclass), family = "binomial",
##     data = passengers, na.action = "na.exclude")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.7303  -0.6780  -0.3953   0.6485   2.4657
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       2.741425   0.266050  10.304  < 2e-16 ***
## age_cent         -0.036985   0.007656  -4.831 1.36e-06 ***
## Sexmale          -2.522781   0.207391 -12.164  < 2e-16 ***
## factor(Pclass)2 -1.309799   0.278066  -4.710 2.47e-06 ***
## factor(Pclass)3 -2.580625   0.281442  -9.169  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 964.52  on 713  degrees of freedom
## Residual deviance: 647.28  on 709  degrees of freedom
##   (177 observations deleted due to missingness)
## AIC: 657.28
##
## Number of Fisher Scoring iterations: 5
```

Tolkning av logit-regresjon

Tolkning av logit-estimater, utover retning, er gankse vanskelig. Heldigvis kan vi regne ut sannsynligheter basert på disse estimatene. Her bruker vi estimatene fra regresjonen til å regne fra logit til sannsynligheter manuelt:

```
# Konstantleddet (kvinne, 28 år, 1 klasse)
exp(2.741425) / (1 + exp(2.741425))
```

```
## [1] 0.9394272
```

```
# Alder (1 enhets økning)
exp(2.741425 + (-0.036985 * 1)) / (1 + exp(2.741425 + (-0.036985 * 1)))
```

```
## [1] 0.9372881
```

```
# Alder (10 enhets økning)
exp(2.741425 + (-0.036985 * 10)) / (1 + exp(2.741425 + (-0.036985 * 10)))
```

```
## [1] 0.9146339
```

```
# Mann, 38 år, 3 klasse
exp(2.741425 + (-0.036985 * 10) + (-2.522781 * 1) + (-2.580625 * 1)) /
  (1 + exp(2.741425 + (-0.036985 * 10) + (-2.522781 * 1) + (-2.580625 * 1)))
```

```
## [1] 0.06112101
```

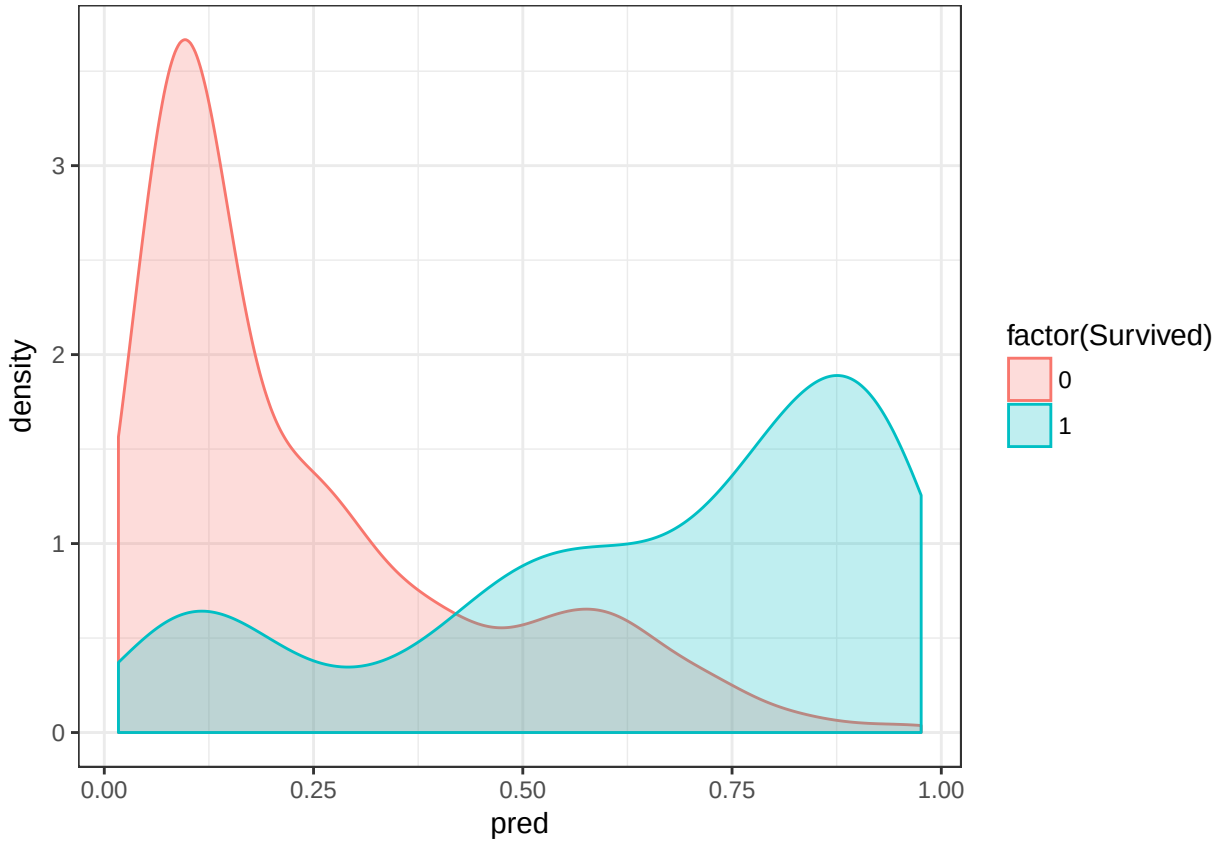
Dette kan også gjøres automatisk. Her med in sample prediksjon:

```
# Logits
predict(pass_logit)

# Sannsynlighet
predict(pass_logit, type = "response")

# Sannsynlighet, med standardfeil
predict(pass_logit, type = "response", se = TRUE)
```

Der er også mange måter å sjekke om denne prediksjonen er god; her er ett forslag:

```
passengers$pred <- predict(pass_logit, type = "response")

ggplot(passengers, aes(x = pred, color = factor(Survived), fill = factor(Survived))) +
  geom_density(alpha = .25) +
  theme_bw()
```

```
## Warning: Removed 177 rows containing non-finite values (stat_density).
```



Neste gang:

- Mer wrangling
- Samspill
- Diagnostisering
- Plotte effekter med multipel regresjon
- Ønsker?

Bonus for LaTeX-elskere:

```
# install.packages("stargazer")
library(stargazer)
```

```
##
## Please cite as:
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer
```

```
stargazer(pass_reg, pass_reg2, pass_reg3, pass_logit,
          star.cutoffs = c(.05, .01, .001),
          column.sep.width = ".01cm",
          no.space = FALSE,
          covariate.labels = c("Alder (sentrert)", "Kjønn (mann)",
                               "Klasse (2)", "Klasse (3)", "Konstantledd"),
          keep.stat = c("n", "rsq", "adj.rsq", "ll"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: on., sep. 12, 2018 - kl. 14.12 +0200

Table 1:

	Dependent variable:			
	Survived			
	OLS			*logistic*
	(1)	(2)	(3)	(4)
Alder (sentrert)	-0.003^{*}	-0.001	-0.005^{***}	-0.037^{***}
	(0.001)	(0.001)	(0.001)	(0.008)
Kjønn (mann)		-0.547^{***}	-0.479^{***}	-2.523^{***}
		(0.032)	(0.031)	(0.207)
Klasse (2)			-0.208^{***}	-1.310^{***}
			(0.042)	(0.278)
Klasse (3)			-0.407^{***}	-2.581^{***}
			(0.038)	(0.281)
Konstantledd	0.411^{***}	0.755^{***}	0.972^{***}	2.741^{***}
	(0.018)	(0.026)	(0.034)	(0.266)
Observations	714	714	714	714
R^2	0.006	0.291	0.390	
Adjusted R^2	0.005	0.289	0.387	
Log Likelihood				-323.642
Note:				*p<0.05; **p<0.01; ***p<0.001