Advanced Dataanalysis and Statistical Modelling: Assignment 2

Spring 22, 02424
April 1, 2022

s170303, Andreas Engly
s174434, Mads Esben Hansen

Danmarks
Tekniske Universitet

Contents

1 Ozone

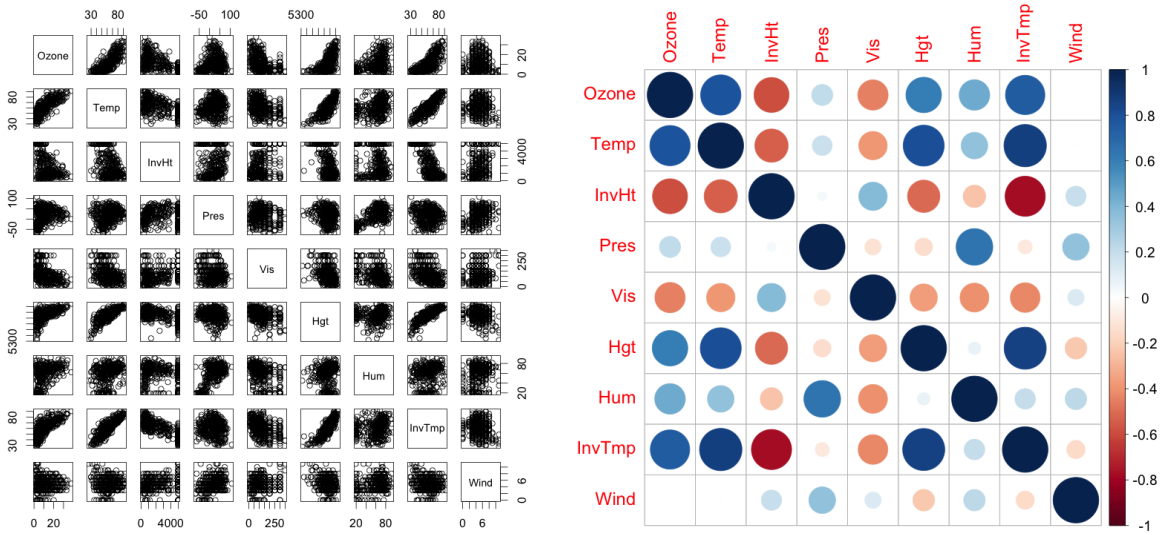
The first part of the report concerns itself with modeling of ozone concentration in Los Angeles. The data used is *ozone* which is included in the R package *gclus*.

Variable	Description	Type
Ozone	The concentration of ozone measured in parts per million (ppm)	Continuous ($\mathbb{R}_+$)
Temp	Temperature measured in fahrenheit (F)	Continuous ($\mathbb{R}$)
InvHt	Inversion base height denotes the height where temperature starts to decrease as height increases. It is measured in feet (ft)	Continuous ($\mathbb{R}$)
Pres	Pressure gradient measured in millimetre of mercury (mmHg)	Continuous ($\mathbb{R}_+$)
Vis	Visibility measured in miles (mi)	Continuous ($\mathbb{R}_+$)
Hgt	Vandenburg 500 millibar (mb) height denotes the height where the air pressure is 500 mb. It is measured in meters (m)	Continuous ($\mathbb{R}$)
Hum	Humidity measured as percentage of water vaper per unit dry air	Continuous ($\mathbb{R}_+$)
InvTmp	Inversion base temperature denotes the temperature at the height where temperature starts to decrease as height increases. It is measured in feet (ft)	Continuous ($\mathbb{R}$)
Wind	Wind speed measured in miles per hour (mph)	Continuous ($\mathbb{R}_+$)

Table 1 – Description of the data set *ozone* from the package *gclus* [1]

1.1

Figure 1 shows an analysis of the dependence between the variables. From Figure 1 (b) it is clear that there is positive correlation between `Ozone` and `Temp`, `Hgt`, `Hum`, and `InvTmp`. It seems there is a negative correlation between `Ozone` and `InvHt`, and `Vis`. The general interpretation of this is that for high `Temp`/`Hgt`/`Hum`/`InvTmp` the ozone level tends to be high and vice versa, whereas the opposite is the case for `InvHt` and `Vis`. From Figure 1 (a) is seems that there might be some non-linear dependence between e.g. `Ozone` and `Hgt`.



(a) Raw variable comparison. **(b)** Correlation between variables.

Figure 1 – Overview of variables and the direct relationship.

1.2

The first thing we will do is to fit a general linear model to the data. For simplicity we only consider first-order effects. For model selection we use *type II* as described in [2], i.e. backward selection. Therefore we start off with a *sufficient* model:

$$\texttt{Ozone} = \theta_0 + \theta_1\texttt{Temp} + \theta_2\texttt{InvHt} + \theta_3\texttt{Pres} + \theta_4\texttt{Vis}... \tag{1}$$

$$+ \theta_5\texttt{Hgt} + \theta_6\texttt{Hum} + \theta_7\texttt{InvTmp} + \theta_8\texttt{Wind} + \varepsilon \tag{2}$$

$\varepsilon \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$. *Type II* model selection dictates that variables should be removed one by one until the model is significantly worse c.f. section 3.6 [2] than the *sufficient* model. We end up with

$$\texttt{Ozone} = \theta_0 + \theta_1\texttt{Temp} + \theta_2\texttt{Hum} + \theta_3\texttt{InvTmp} + \varepsilon \tag{3}$$

$\varepsilon \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$. The parameter estimates are given in Table 2.

Variable	Mean	SD
θ_0	$-1.049 \cdot 10^{1}$	$1.616 \cdot 10^{0}$
θ_1	$3.296 \cdot 10^{-1}$	$2.109 \cdot 10^{-2}$
θ_2	$7.738 \cdot 10^{-2}$	$1.339 \cdot 10^{-2}$
θ_3	$-1.004 \cdot 10^{-3}$	$1.639 \cdot 10^{-4}$
σ^2	4.524^2	NA

Table 2 – Estimated parameters for the general linear model from Equation 3.

Figure 2 shows a rudimentary residual analysis of the general linear model. The figures display one major issues: The variance of the residuals seem to increase with the fitted values.

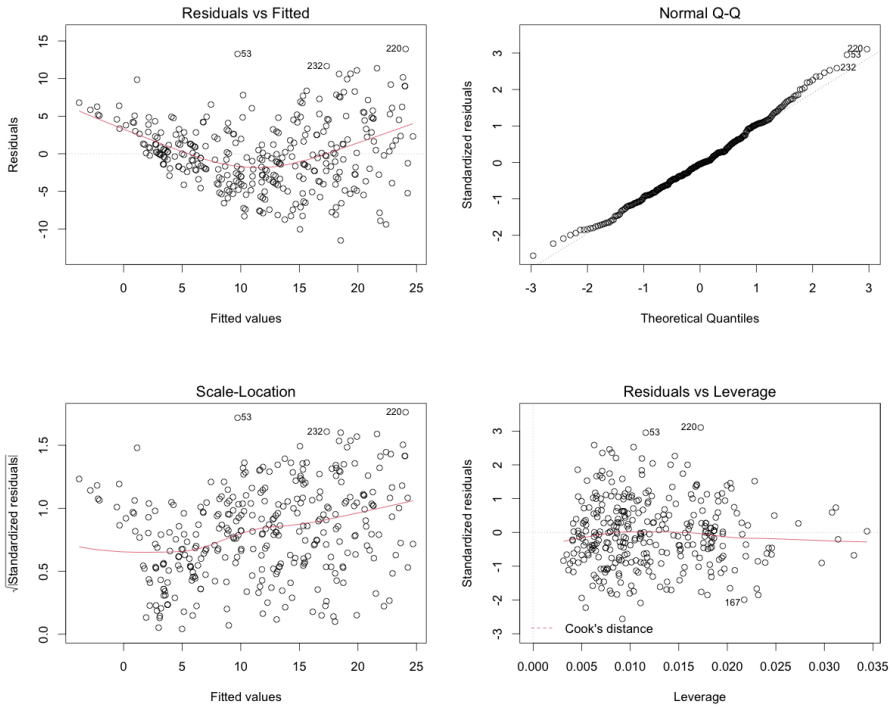


Figure 2 – Residual analysis of general linear model from Equation 3.

Figure 3 shows a the relationship between predictors and residuals. Upon visual inspection it seems there is a non-linear relationship between especially temperature and residuals. This together with the heteroscedasticity we saw in Figure 2 implies a potential need for transformation of data and/or use of generalized linear models.

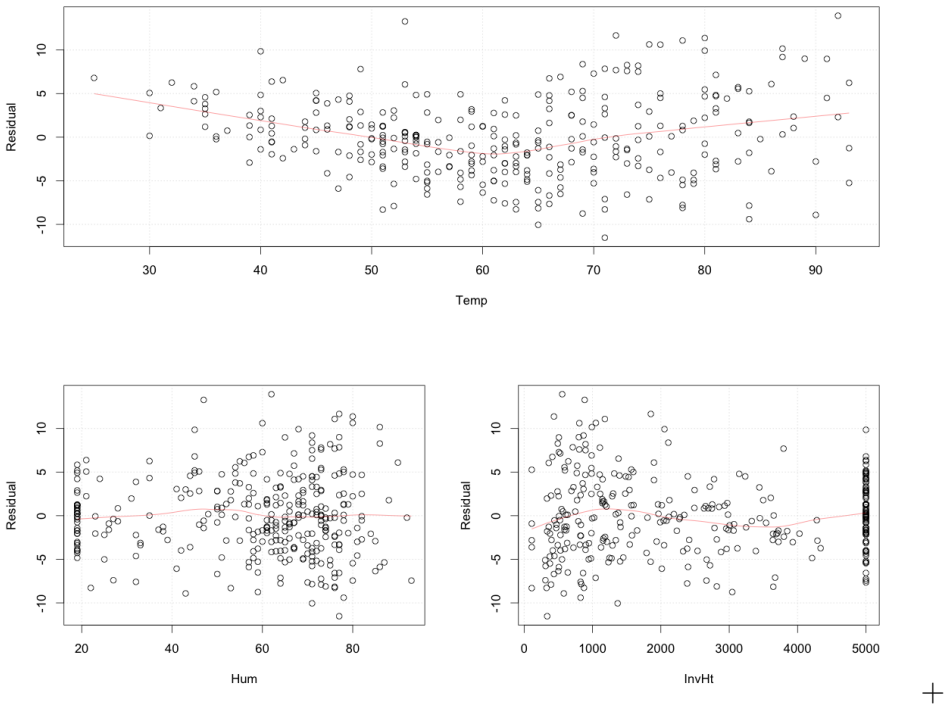


Figure 3 – Relationship between predictors and residuals for the general linear model from Equation 3.

1.3

Now we will try to fit a generalized linear model instead of a general linear model.

The concentration of ozone can obviously not be negative. We can either choose to consider the ozone concentration as a genuine concentration and assume it to be continuous. We can also choose to view is as the number of particles in a given sample, i.e. count data. By rule of thumb we expect it to follow a Gamma or with a little bit of imagination a Poisson distribution cf [2] p. 89. The heteroscedasticity seen in 1.1 implies we need a non-linear link function, e.g. *log*, *sqrt* or similar. For general linear model model selection is done by means of an F-test. For generalized linear models we are dealing with different models with potentially the same number of parameters. This complicates the *expectation* to the residuals (remembering the SSE follows a χ^2 distribution) and comparison thereof. Instead we will turn to more generic methods of comparison, i.e. information criteria. In particular we will emphasize Akaike information criterion (AIC), but also look at the bayesian information criterion (BIC). For model reduction we use a χ^2-test cf [2] section 4.6 to find the model that minimizes AIC.
Table 3 shows the information criteria and goodness of fit (GOF) test (p-value based on deviance residuals) of a number of generalized linear models along with the general linear model from the previous section. Additionally, Gaussian distributions with log and sqrt as link functions are also present, both of which could have been obtained by transformation and using a general linear model. Notice all generalized models have better statistics than the original general linear model. It seems either a Gamma or Poisson distribution with a log link function is the best choice for our data. Based on GOF, it seems that using a Poisson distribution will underestimate the variance, therefore Gamma with log

link function is preferred. The reduced models have all been found by *type II* selection. The Gamma model with log link function is presented in section 1.4.

Density	**Link**	**AIC**	**BIC**	**GOF-test**
Gamma	Inverse	1847.6	1870.4	0.2033
Gamma	Log	1802.0	1824.8	0.1898
Gamma	Sqrt	1835.4	1854.4	0.1156
Poisson	Log	1800.2	1819.2	$9.951 \cdot 10^{6}$
Poisson	Sqrt	1834.0	1853.0	$5.269 \cdot 10^{-8}$
Poisson	Inverse	1881.1	1903.9	$1.159 \cdot 10^{-11}$
Gaussian	Identity	1938.7	1957.7	0.4896
Gaussian	Log	1875.4	1898.2	0.4897
Gaussian	Sqrt	1885.8	1912.4	0.4896

Table 3 – Information criteria for the generalized linear models. Last model is the model from section 1.1.

Figure 4 shows a rudimentary residual analysis of the generalized linear model. Before we saw the variance of the residuals increase with the fitted values. This seems no longer to be the case. Besides a few individual outliers both the Pearson and deviance residuals seem to be OK, i.e. homoscedastic and approximately normal.

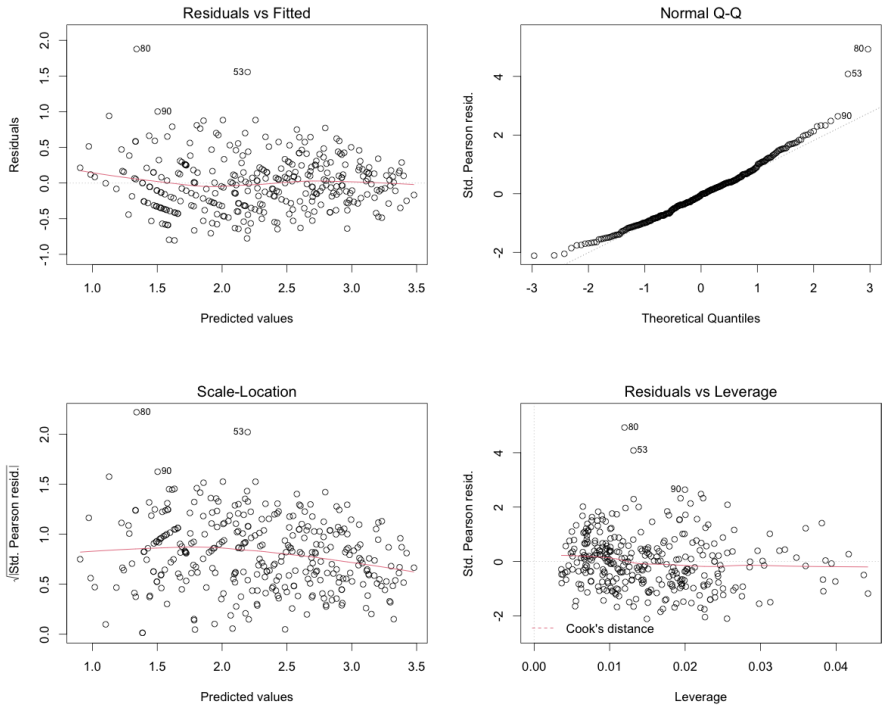


Figure 4 – Residual analysis of general linear model from Equation 5.

Figure 5 shows a the relationship between predictors and deviance residuals. Upon visual inspection it seems the non-linear relationship is gone. This together with the homoscedasticity we saw in Figure 4 implies a good model.

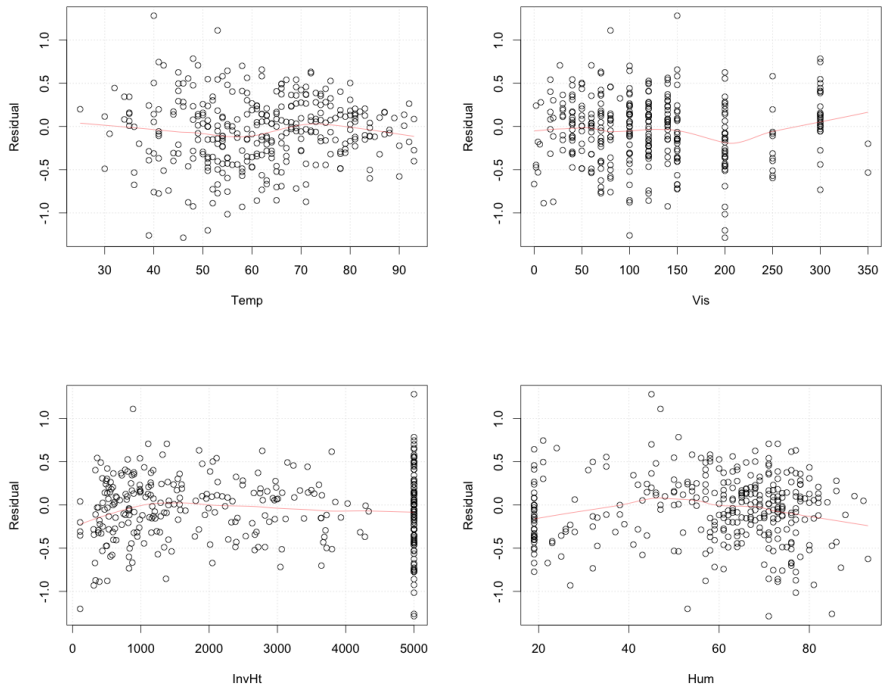


Figure 5 – Relationship between predictors and residuals (deviance) for the generalized linear model.

1.4

We now want to compare the general linear model to the generalized linear model. Comparison of these is not trivially done using an F-test since they have the same number of parameters and are fitted in different domains. Instead we will once again turn to AIC and BIC. Per Table 3 the general linear model has an AIC of 1938.7 and BIC of 1957.7 whereas the generalized linear has 1802.0 and 1824.8, respectively. Based on this alone, it seems the generalized linear model represents the data better then the general linear model. Figures 6 and 7 show the ozone concentration over Los Angeles as a function of temperature together with the fitted values for the general and generalized linear model. It is clear that the general linear model does not capture the non-linear behaviour of the data. It seems to be baised for especially high temperature and underestimate the variance at these temperature.
In total, all evidence both qualitative and quantitative points towards using the generalized linear model.

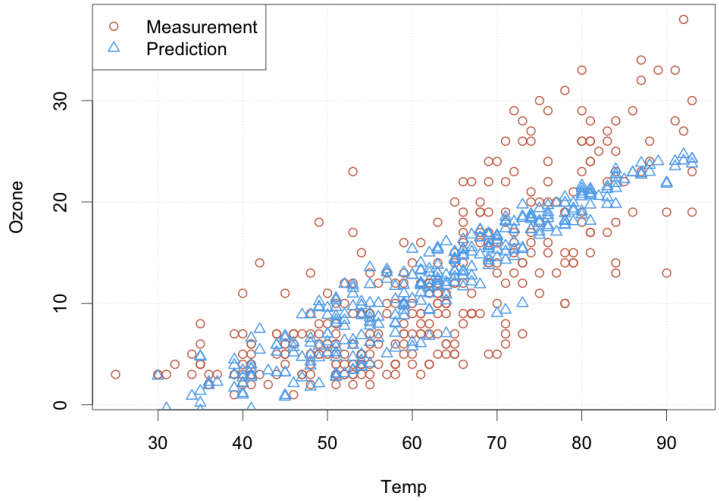


Figure 6 – Ozone measurements and fitted values as function of temperature using the general linear model.

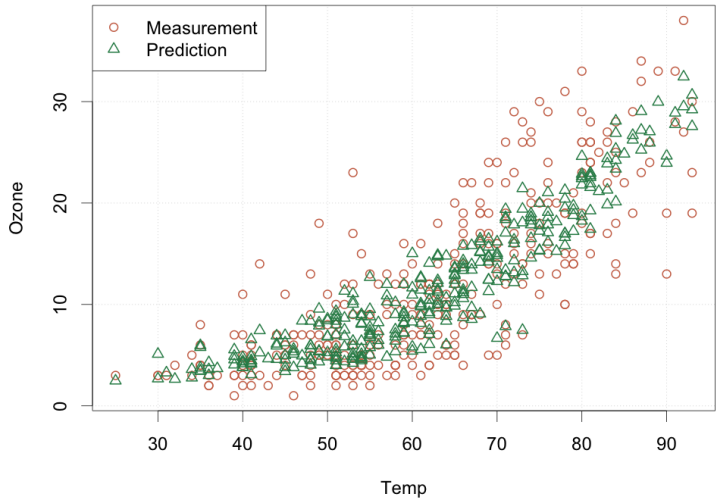


Figure 7 – Ozone measurements and fitted values as function of temperature using the generalized linear model.

1.5

We settled on a generalized linear model with `Temp`, `InvHt`, `Vis`, and `Hum` as predictors, and a log link function

$$\mathbb{E}[\log(\texttt{Ozone})] = \theta_0 + \theta_1\texttt{Temp} + \theta_2\texttt{InvHt} + \theta_3\texttt{Vis} + \theta_4\texttt{Hum} \tag{4}$$

$$\mathbb{E}[\texttt{Ozone}] = \exp\left[\theta_0 + \theta_1\texttt{Temp} + \theta_2\texttt{InvHt} + \theta_3\texttt{Vis} + \theta_4\texttt{Hum}\right] \tag{5}$$

Table 4 shows the parameter estimates of the generalized linear model. Figure 7 shows fitted values using the model. It seems to follow the data nicely.

Variable	**Mean**	**SD**
θ_0	$6.215 \cdot 10^{-1}$	$1.537 \cdot 10^{-1}$
θ_1	$2.700 \cdot 10^{-2}$	$1.812 \cdot 10^{-3}$
θ_2	$-1.096 \cdot 10^{-4}$	$1.424 \cdot 10^{-5}$
θ_3	$-6.050 \cdot 10^{-4}$	$3.116 \cdot 10^{-4}$
θ_4	$6.176 \cdot 10^{-3}$	$1.191 \cdot 10^{-3}$

Table 4 – Parameter estimates for generalized linear model from Equation 5.

2 Clothing

In this section, we wish to model the number of times a person changes clothing throughout a day. From the exponential dispersion family, will use the binomial distribution and poisson distribution respectively. This means they can both be written in the following form

$$f_Y(y;\theta) = c(y,\lambda)\ \exp(\lambda\{\theta y - \kappa(\theta)\}), \quad \theta \in \Omega \tag{6}$$

2.1

We wish to model the number of times that an individual changes clothes during a day. As each observations corresponds to a Bernoulli trial, it is reasonable to model the response variable as binomially distributed with parameters $p_i \in \{0,..,1\}$ and number of observations $n_i \in \mathbb{N}$.

$$Y_i|\boldsymbol{x}_i \sim \text{Binom}(n_i, p_i) \tag{7}$$

We will use all available data and furthermore add mixed terms. The corresponding coefficients are seen from Table 5.

Variable	**Coefficient**
intercept	β_0
time	β_1
nobs	β_2
sex	β_3
tOut	β_4
tInOp	β_5
tOut:sexmale	β_6
sexmale:tInOp	β_7

Table 5 – Description of the data

What we believe to be the *sufficient model* is thus given by 8 coefficients. The *linear predictor* is

$$\begin{aligned} g(\mu_i) &= \eta_i \\ &= \beta_0 + x_{i,1}\beta_1 + .. + x_{i,7}\beta_7 \\ &= (\boldsymbol{x}_i)^T\boldsymbol{\beta} \end{aligned} \tag{8}$$

The relationship between the mean value parameter and the linear predictor is according to definition 4.5 in [2] given by the *canonical link function*

$$g(\mu_i) = \log(\frac{\mu_i}{1-\mu_i}) \tag{9}$$

which is known as the *logit*. It connects the two parameterizations. The inverse mapping is denoted the *logistic function* and is given by

$$\begin{aligned}\mu_i &= g^{-1}(\eta_i) \\ &= \frac{\exp(\eta)}{1+\exp(\eta)} \\ &= p_i \in \{0, .., 1\}\end{aligned}$$

The *canonical link function* is not necessarily the best suited for the model.
The above model formulation leads to a model with the following output table.

```
Call:
glm(formula = cbind(clo, nobs - clo) ~ time + nobs + tOut * sex +
    tInOp * sex, family = binomial(link = "logit"), data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5181  -1.1287  -0.6539   0.3773   2.9102

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -6.24445    5.09241  -1.226   0.2201
time           0.58359    0.62886   0.928   0.3534
nobs          -0.43417    0.68520  -0.634   0.5263
tOut          -0.03069    0.04529  -0.678   0.4981
sexmale       10.86588    6.06686   1.791   0.0733 .
tInOp          0.13269    0.15310   0.867   0.3861
tOut:sexmale   0.05233    0.07847   0.667   0.5049
sexmale:tInOp -0.49468    0.24456  -2.023   0.0431 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 193.66  on 135  degrees of freedom
Residual deviance: 167.40  on 128  degrees of freedom
AIC: 280.93

Number of Fisher Scoring iterations: 5
```

Figure 8 – Summary of model with logit link function

As seen the residual deviance is 167.40 on 128 degrees of freedom. The goodness of fit statistic can then be tested

$$\mathbb{P}[D(\boldsymbol{\mu}(\hat{\boldsymbol{\beta}})) \leq \chi(128)^2] = 0.011 \tag{10}$$

This suggests that the assumption of the binomial distribution is problematic as the test statistic is extreme. This might indicate the presence of overdispersion. For now this consideration is ignored. It is seen that only sexmale:tInOp is significant. By formulating successive tests corresponding to a chain of hypothesis, we will attempt to reduce the model. Using a *type II* reduction we end up with the following model.

```
Call:
glm(formula = cbind(clo, nobs - clo) ~ sex, family = binomial(link = "logit"),
    data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3735  -0.7849  -0.7849   0.1631   3.0756

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -1.5721     0.1419 -11.081  < 2e-16 ***
sexmale      -1.1839     0.2759  -4.292 1.77e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 193.66  on 135  degrees of freedom
Residual deviance: 172.57  on 134  degrees of freedom
AIC: 274.1

Number of Fisher Scoring iterations: 5
```

Figure 9 – Summary of final model.

Checking the test statistic yields

$$\mathbb{P}[D(\boldsymbol{\mu}(\hat{\boldsymbol{\beta}})) \leq \chi(134)^2] = 0.0134 \tag{11}$$

which is still significant for significance levels under 5%. The interpretation of the model is that males tend to change clothes less often than women. This can be seen as the *sexmale* coefficient is negative leading to a smaller linear predictor. As the logistic function is monotone, this results in a smaller parameter p. However, as already stated the usefulness is very limited as it might suffer from overdispersion. To see if this could be the case, a boxplot is made.

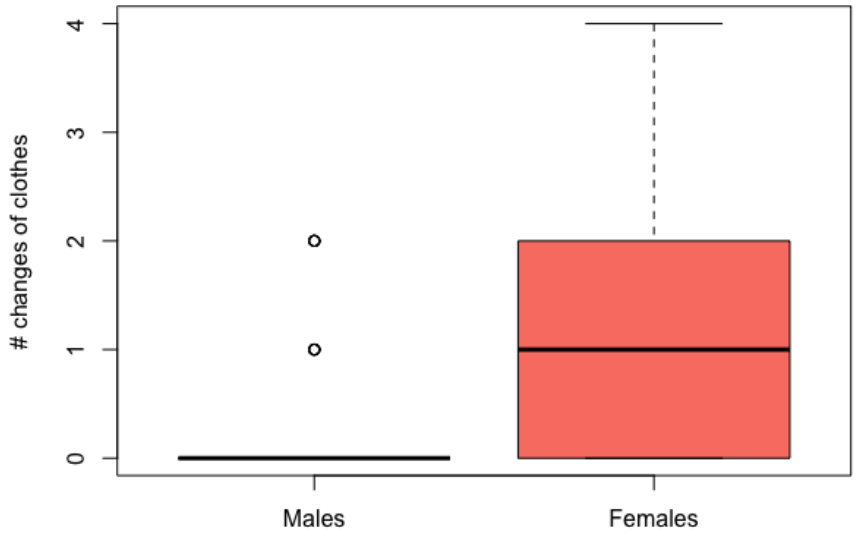


Figure 10 – Comparison between males and females.

It is seen that the variance differs a lot between the groups. While the model can handle heteroscedasticity, it does not guarantee that the variance can be captured to a satisfactory degree in the model. A summary of the generalized linear model with the different link functions is found below.

Link function	**Goodness of fit**	**AIC**
logit	0.01385	274.1
probit	0.01385	274.1
cloglog	0.01385	274.1

Table 6 – Summary of different link functions.

As seen from Table 6, the resulting models have the same goodness of fit and AIC. This is because the model parameter has no dependency on any input variable apart from the gender variable. Transforming this back into the original domain from the link domain will thus result in the same probability parameter.

2.2

Instead of assuming a binomial distribution, we now attempt to formulate a model where the response variable is assumed to follow a poisson distribution.

$$Y_i|\boldsymbol{x}_i \sim \text{Pois}(\lambda_i) \tag{12}$$

The *linear predictor* maintains the same structure as in Equation 8, but the *canonical link function* is now

$$g(\lambda_i) = \log(\lambda_i) \tag{13}$$

The inverse is then given by the exponential function ensuring the parameter range is in correspondence with the rate parameter in the Poisson distribution.

$$\begin{aligned}\lambda_i &= g^{-1}(\eta_i) \\ &= \exp((\boldsymbol{x}_i)^T\boldsymbol{\beta}) \in \mathbb{R}_+\end{aligned}$$

The summary of the initial model is

```
Call:
glm(formula = clo ~ time + nobs + tOut * sex + tInOp * sex, family = poisson(link = "log"),
    data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.4486  -1.0964  -0.6486   0.3464   2.2803

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -5.59587    4.66452  -1.200   0.2303
time           0.54345    0.57886   0.939   0.3478
nobs          -0.13075    0.63219  -0.207   0.8361
tOut          -0.02595    0.04131  -0.628   0.5299
sexmale        9.82630    5.65449   1.738   0.0822 .
tInOp          0.11126    0.13996   0.795   0.4266
tOut:sexmale   0.04605    0.07389   0.623   0.5331
sexmale:tInOp -0.44604    0.22818  -1.955   0.0506 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 170.05  on 135  degrees of freedom
Residual deviance: 145.01  on 128  degrees of freedom
AIC: 278

Number of Fisher Scoring iterations: 6
```

Figure 11 – Summary of initial model with log link function

The probability of observing a more extreme test statistic is

$$\mathbb{P}[D(\boldsymbol{\mu}(\hat{\boldsymbol{\beta}})) \leq \chi(128)^2] = 0.144 \tag{14}$$

which indicates that the model is sufficient. Proceeding with the same reduction scheme as for the binomial case, we arrive at the following model.

```
Call:
glm(formula = clo ~ sex, family = poisson(link = "log"), data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3093  -0.7588  -0.7588   0.1503   2.4572

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.1542     0.1291  -1.194    0.232
sexmale      -1.0911     0.2632  -4.145  3.4e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 170.05  on 135  degrees of freedom
Residual deviance: 150.04  on 134  degrees of freedom
AIC: 271.03

Number of Fisher Scoring iterations: 6
```

Figure 12 – Summary of final model without offset

The residual deviance yields the following p-value

$$\mathbb{P}[D(\boldsymbol{\mu}(\hat{\boldsymbol{\beta}})) \leq \chi(134)^2] = 0.163 \tag{15}$$

indicating that the deviance can be explained by the χ^2-distribution. As in the binomial case, the model indicates that men are less likely to change clothes throughout the day. As rates are expressed in a time unit, we can possibly improve the model by including an offset.

$$\begin{aligned} \log(\lambda_i) &= \log(\text{time}_i) + (\boldsymbol{x}_i)^T\boldsymbol{\beta} \\ &\Leftrightarrow \\ \log(\frac{\lambda_i}{\text{time}_i}) &= (\boldsymbol{x}_i)^T\boldsymbol{\beta} \\ &\Leftrightarrow \\ \frac{\lambda_i}{\text{time}_i} &= e^{(\boldsymbol{x}_i)^T\boldsymbol{\beta}} \end{aligned}$$

As seen this corresponds to scaling the parameter to the appropriate time unit. Starting off with the same sufficient model yields the following final model.

```
Call:
glm(formula = clo ~ sex + offset(time), family = poisson(link = "log"),
    data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6259  -0.8498  -0.7620   0.4821   2.4361

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -6.8710     0.1291  -53.22   <2e-16 ***
sexmale      -1.0240     0.2632   -3.89    1e-04 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 167.53  on 135  degrees of freedom
Residual deviance: 150.02  on 134  degrees of freedom
AIC: 271.01

Number of Fisher Scoring iterations: 6
```

Figure 13 – Summary of final model with offset

Including the offset only changes the AIC of the final model marginally.

Description	Goodness of fit	AIC
log	0.163	271.03
log with offset	0.162	271.01

Table 7 – Summary of Poisson.

2.3

As stated in the subsections above, both models indicate that men are less likely to change clothes throughout the day. The coefficient β_{sexmale} is negative and significant in both cases. As this is tested against the restricted model $\mathcal{H}_A : \beta_{\text{sexmale}} = 0$, it cannot be rejected that there is a significant difference between males and females on this matter. As the poisson assumption leads to a model where the deviance can be explained (and further have a lower AIC), it is to be preferred over the binomial.

Under the binomial distribution, the parameter p_i corresponds to the probability of changing clothes at each observation. Transformation the coefficient back from the link domain yields 17.2 pct. for women and 6.0 pct. for men. Under the poisson distribution, the parameter corresponds to the daily change rate of 0.857 for women and 0.288 for men.

3 Fan speed

The third part of the report concerns itself with perceived sensation for different settings of a fan. The data used is *CeilingFan.csv* and is supplied for this assignment. Table 8 contains a descriptions of the variables and the meaning.

Variable	**Description**	**Type**
subjId	Unique identifier for a test person	Factor ({0,...,21})
fanSpeed	Speed of fan	Integer ({0,1,2})
TSV	Thermal sensation vote	Ordinal ({0,1,2})
fanType	Type of fan exposure	Factor ({downstream, upstream})

Table 8 – Description of the data set *CeilingFan.csv*.

3.1

We start by addressing the relationship between `TSV` and `fanSpeed`. We want to know if we can assume that the measured `TSV` depends on the `fanSpeed` or not. To answer this question, we perform an ordinary contingency test as described in [3] section 7.5. Table 9 shows the contingency table in question, Figure 14 shows the percentage distribution of `TSV` as a function of `fanSpeed`.

fanSpeed \ TSV	**0**	**1**	**2**	**Total**
0	97	40	8	145
1	20	24	8	52
2	20	10	10	40
Total	137	74	26	237

Table 9 – Contingency table for `TSV` and `fanSpeed`.

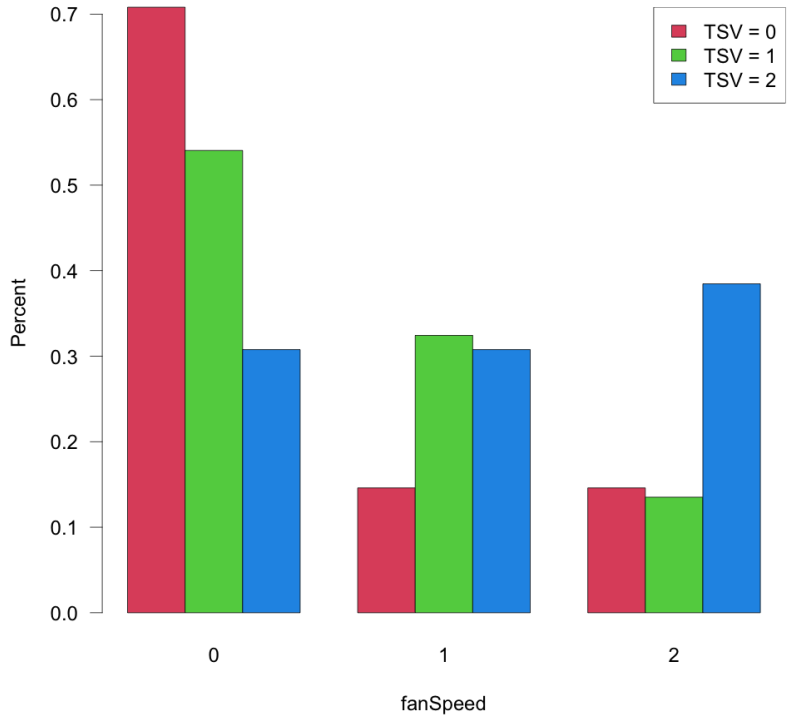


Figure 14 – Overview of distribution of `TSV` as a function of `fanSpeed`. Percentage given is percentage of `TSV` such that $p_{TSV=0|fanSpeed=0} + p_{TSV=1|fanSpeed=0} + p_{TSV=2|fanSpeed=0} = 0$ etc.

We formulate the null-hypothesis

$$H_0 : \quad p_{i0} = p_{i1} = p_{i2} \tag{16}$$

i.e. the probability of obtaining an outcome in a row category does not depend on the given column. We get a χ^2 test statistic of 22.715. On 4 degrees of freedom, this yields a p-value of $1.44 \cdot 10^{-4}$. Hence we reject the null-hypothesis that `fanSpeed` does not have an impact on `TSV`.

3.2

The R package *ordinal* contains Cumulative Link Models (CLM) which are regression models where the response should be a factor, which will be interpreted as an ordinal response with levels ordered as in the factor. Firstly we consider the *null* model, i.e. the model that assumes no relationship between `fanSpeed` and `TSV`. In R we do this by

```
m1.p <- clm( TSV ~ 1, data = fan.dat ).
```

We now want a model that uses `fanSpeed` as a predictor to predict the probability of any `TSV`, we do this by

```
m2.p <- clm( TSV ~ fanSpeed, data = fan.dat ).
```

Afterwards we are able to compare the two models using *anova* as requested. Doing so yields a significant difference with a p-value of $3.6 \cdot 10^{-4}$, therefore we reject the null hypothesis that there is independence between `TSV` and `fanSpeed`.

3.3

We now want to develop a model for `TSV` as a function of `fanSpeed` and `fanType`. Example 4.12 in [2] shows a similar situation. Here a logistic model for the cumulative probabilities is used. As it turns out this is exactly how *clm* from the *ordinal* package works. We consider the cumulative probabilities

$$\Pi_{i,j} = \sum_{j=0}^{2} p_{i,j}, \quad i \in \{upstream,\ downstream\} \tag{17}$$

As in the example, *clm* use a *logit* transformation of the cumulative probabilities as a default. However, one could also use e.g. *probit* or *cloglog*. Table 10 shows AIC for different link functions. *Type II* model selection was used for all model.

Link function	**AIC**
logit	421.5
probit	420.1
cloglog	417.3

Table 10 – Information criterion of the cumulative probability model using different transformations (link functions).

Since *cloglog* has the lowest AIC, this is our preferred model. The model is given by

$$\Pi_{i,j} = 1 - \exp\left(-\exp\left(\alpha_{i,j} + \beta_j \cdot \texttt{fanSpeed}\right)\right), \quad i \in \{upstream,\ downstream\}. \tag{18}$$

3.4

Table 11 shows the parameter estimates for the model. Notice the parameters β_U and β_D, which are the upstream and downstream linear coefficients, are both positive. This means that as `fanSpeed` becomes larger, `TSV` will go towards the last categories, i.e. `TSV`=2. Also notice that $\beta_U > \beta_D$ meaning that this transition of probability happens faster for upstream than downstream. The α estimates are estimates of probabilities when `fanSpeed`=0.

Parameter	**Estimate**
$\alpha_{U,0}$	0.577
$\alpha_{U,1}$	0.331
$\alpha_{U,2}$	0.092
$\alpha_{D,0}$	0.709
$\alpha_{D,1}$	0.245
$\alpha_{D,2}$	0.046
β_U	0.511
β_D	0.125

Table 11 – Parameters of model given in Equation 18.

Figure 15 shows the probabilities of `TSV` as a function of `fanSpeed`. Notice that `fanSpeed` has been extrapolated beyond the known capabilities of the fan. This is only done for pedagogical purposes and to better show the dynamics of the model. From the figure, it is quite clear that whenever the fan speed is increase, probabilities transitions to a higher TSV value.

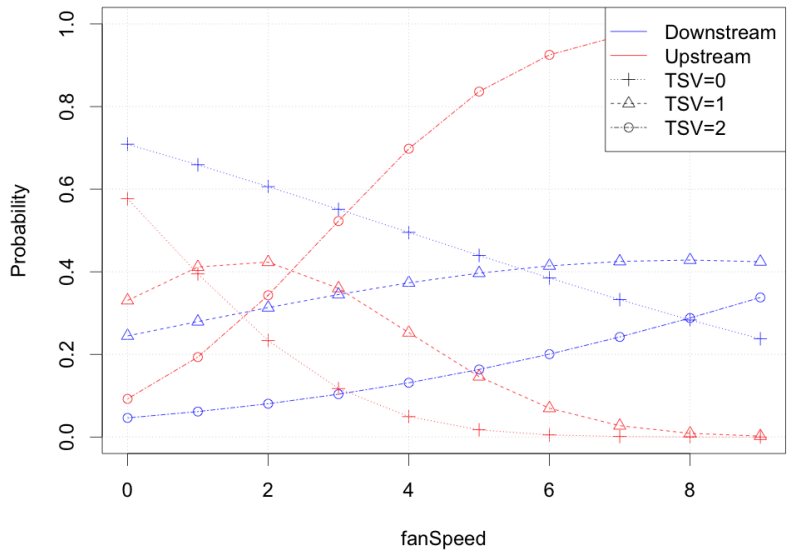


Figure 15 – Modelled `TSV` as a function of `fanSpeed`. Notice the fan speed is extrapolated beyond the capabilities of the fan.

References

[1] CRAN. Package 'gclus'. [Online]. Available: https://cran.r-project.org/web/packages/gclus/gclus.pdf

[2] H. Madsen and P. Thyregod, *Introduction to General and Generalized Linear Models*, 2010.

[3] P. B. Brockhoff, J. K. Møller, E. W. Andersen, P. Bacher, and L. E. Christiansen, *Introduction to statistics at DTU*, 2017.

A R-code

```
setwd( '/Users/mads/Google␣Drev/Skole/Uni/10_semester/02424/Assignment2'
    ↪ )

#### Part 1 ----
library("gclus")
data(ozone)
head(ozone)

## 1)
png("figures/cor-raw.png", width = 800, height = 800, pointsize = 24)
pairs(ozone)
dev.off()

require(corrplot)
png("figures/cor-heat.png", width = 800, height = 800, pointsize = 24)
corrplot(cor(ozone))
dev.off()


## 2)
names(ozone)
fit1 <- lm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp + Wind,
    ↪  data = ozone)
summary(fit1)
drop1(fit1, test = "F")
fit1 <- update(fit1, .~.-Pres)
drop1(fit1, test = "F")
fit1 <- update(fit1, .~.-Wind)
drop1(fit1, test = "F")
fit1 <- update(fit1, .~.-Hgt)
drop1(fit1, test = "F")
fit1 <- update(fit1, .~.-InvTmp)
drop1(fit1, test = "F")
fit1 <- update(fit1, .~.-Vis)
drop1(fit1, test = "F")
summary(fit1)


png("figures/lm1_res.png", height = 1000, width = 1200, pointsize = 20)
par(mfrow = c(2,2))
plot(fit1)
dev.off()

png("figures/lm1_res2.png", height = 1000, width = 1200, pointsize = 20)
layout(matrix(c(1,2,1,3),2,2))
plot(residuals(fit1) ~ ozone$Temp, xlab = "Temp", ylab = "Residual")
```

```
lines(seq(min(ozone$Temp), max(ozone$Temp), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$Temp, span = 0.5, degree = 1),
         ↪  seq(min(ozone$Temp), max(ozone$Temp), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
plot(residuals(fit1) ~ ozone$Hum, xlab = "Hum", ylab = "Residual")
lines(seq(min(ozone$Hum), max(ozone$Hum), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$Hum, span = 0.5, degree = 1),
         ↪ seq(min(ozone$Hum), max(ozone$Hum), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
plot(residuals(fit1) ~ ozone$InvHt, xlab = "InvHt", ylab = "Residual")
lines(seq(min(ozone$InvHt), max(ozone$InvHt), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$InvHt, span = 0.5, degree = 1)
         ↪ , seq(min(ozone$InvHt), max(ozone$InvHt), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
dev.off()


## 3)
## Gamma cano. (inverse)
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
   ↪ Wind, data = ozone,family = Gamma(link = "inverse"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-InvTmp)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Vis)
drop1(fit1.glm)
AIC(fit1.glm)
BIC(fit1.glm)

fit1.glm
summary(fit1.glm)
(pval <- 1 - pchisq(58.462/(0.1689992), 325)) # residual deviance,
   ↪ corresponding df


## Gamma log
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
   ↪ Wind, data = ozone,family = Gamma(link = "log"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
```

```
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-InvTmp)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm)
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(51.091/(0.14714), 325)) # residual deviance,
   ↪ corresponding df
fit2.glm = fit1.glm

## Gamma sqrt
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
   ↪ Wind, data = ozone,family = Gamma(link = "sqrt"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-InvTmp)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Vis)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(56.727/(0.1589725), 326)) # residual deviance,
   ↪ corresponding df

## possion cano (log)
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
   ↪ Wind, data = ozone,family = poisson(link = "log"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-InvTmp)
```

```
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
#fit2.glm = fit1.glm
summary(fit1.glm)
(pval <- 1 - pchisq(445.42, 325))



## possion sqrt
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
    ↪ Wind, data = ozone,family = poisson(link = "sqrt"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-InvTmp)
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(479.2, 325))


## possion inverse
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
    ↪ Wind, data = ozone,family = poisson(link = "inverse"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Vis)
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(524.29, 324))


## Gassian
summary(glm(Ozone ~ Temp + InvHt + Hum, data = ozone, family = gaussian))
(pval <- 1 - pchisq(6673.1/20.46953, 326))
```

```
fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
    ↪ Wind, data = ozone,family = gaussian(link = "log"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Wind)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Vis)
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(5475.1/16.8468, 325))

fit1.glm <- glm(Ozone ~ Temp + InvHt + Pres + Vis + Hgt + Hum + InvTmp +
    ↪ Wind, data = ozone,family = gaussian(link = "sqrt"))
summary(fit1.glm)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Hgt)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Pres)
drop1(fit1.glm, test = "Chisq")
fit1.glm <- update(fit1.glm, .~.-Vis)
drop1(fit1.glm, test = "Chisq")
AIC(fit1.glm)
BIC(fit1.glm)
summary(fit1.glm)
(pval <- 1 - pchisq(5615.3/17.33122, 324))


## plots
png("figures/glm1_res.png", height = 1000, width = 1200, pointsize = 20)
par(mfrow = c(2,2))
plot(fit2.glm)
dev.off()

png("figures/glm1_res2.png", height = 1000, width = 1200, pointsize = 20)
layout(matrix(c(1,2,3,4),2,2))
plot(residuals(fit2.glm) ~ ozone$Temp, xlab = "Temp", ylab = "Residual")
lines(seq(min(ozone$Temp), max(ozone$Temp), length.out = 100),
     predict(loess(residuals(fit2.glm) ~ ozone$Temp, span = 0.5, degree =
        ↪  1), seq(min(ozone$Temp), max(ozone$Temp), length.out = 100)),
```

```
     col = "red", lwd = 0.5)
grid()
plot(residuals(fit2.glm) ~ ozone$InvHt, xlab = "InvHt", ylab = "Residual"
   ↪ )
lines(seq(min(ozone$InvHt), max(ozone$InvHt), length.out = 100),
     predict(loess(residuals(fit2.glm) ~ ozone$InvHt, span = 0.5, degree
        ↪ = 1), seq(min(ozone$InvHt), max(ozone$InvHt), length.out =
        ↪ 100)),
     col = "red", lwd = 0.5)
grid()
plot(residuals(fit2.glm) ~ ozone$Vis, xlab = "Vis", ylab = "Residual")
lines(seq(min(ozone$Vis), max(ozone$Vis), length.out = 100),
     predict(loess(residuals(fit2.glm) ~ ozone$Vis, span = 0.5, degree =
        ↪ 1), seq(min(ozone$Vis), max(ozone$Vis), length.out = 100)),
     col = "red", lwd = 0.5)
grid()
plot(residuals(fit2.glm) ~ ozone$Hum, xlab = "Hum", ylab = "Residual")
lines(seq(min(ozone$Hum), max(ozone$Hum), length.out = 100),
     predict(loess(residuals(fit2.glm) ~ ozone$Hum, span = 0.5, degree =
        ↪ 1), seq(min(ozone$Hum), max(ozone$Hum), length.out = 100)),
     col = "red", lwd = 0.5)
grid()
dev.off()



## 4+5)
png("figures/lm1.png", height = 800, width = 1000, pointsize = 24)
plot(ozone$Temp, ozone$Ozone, col = "salmon3", lwd = 2,
    xlab = "Temp", ylab = "Ozone")
points(ozone$Temp, fit1$fitted.values, pch = 2, col = "steelblue2", lwd =
   ↪  2)
grid()
legend("topleft", c("Measurement","Prediction"), pch = c(1,2), col = c("
   ↪ salmon3","steelblue2"), lwd = 2, lty = -1)
dev.off()

png("figures/glm1.png", height = 800, width = 1000, pointsize = 24)
plot(ozone$Temp, ozone$Ozone, col = "salmon3", lwd = 2,
    xlab = "Temp", ylab = "Ozone")
points(ozone$Temp, fit2.glm$fitted.values, pch = 2, col = "seagreen", lwd
   ↪  = 2)
grid()
legend("topleft", c("Measurement","Prediction"), pch = c(1,2), col = c("
   ↪ salmon3","seagreen"), lwd = 2, lty = -1)
dev.off()
summary(fit2.glm)
```

```
################## PART 2 | Clothing insulation level: Count data
    ↪ ###################

################## BINOMIAL ###################

setwd("/Users/andreasengly/Documents/Danmarks␣Tekniske␣Universitet/2.␣
    ↪ semester/02424␣Videregende␣dataanalyse/Assignments/Assignment␣2")
data <- read.csv('./dat_count.csv', sep = ";")

model1.logit <- glm(cbind(clo, nobs-clo) ~ time + nobs + tOut*sex + tInOp
    ↪ *sex, data = data, family = binomial(link = "logit"))
model1.probit <- glm(cbind(clo, nobs-clo) ~ time + nobs + tOut*sex +
    ↪ tInOp*sex, data = data, family = binomial(link = "probit"))
model1.cloglog <- glm(cbind(clo, nobs-clo) ~ time + nobs + tOut*sex +
    ↪ tInOp*sex, data = data, family = binomial(link = "cloglog"))

print(paste("The␣p-value␣is:␣", round(1 - pchisq(model1.logit$deviance,
    ↪ model1.logit$df.residual), 5)))
print(paste("The␣p-value␣is:␣", round(1 - pchisq(model1.probit$deviance,
    ↪ model1.probit$df.residual), 5)))
print(paste("The␣p-value␣is:␣", round(1 - pchisq(model1.cloglog$deviance,
    ↪  model1.cloglog$df.residual), 5)))

### 1. selection

drop1(model1.logit, test='Chisq')
model2.logit <- update(model1.logit, .~.-nobs)

drop1(model1.probit, test='Chisq')
model2.probit <- update(model1.probit, .~.-nobs)

drop1(model1.cloglog, test='Chisq')
model2.cloglog <- update(model1.cloglog, .~.-nobs)

### 2. selection

drop1(model2.logit, test='Chisq')
model3.logit <- update(model2.logit, .~.-tOut:sex)

drop1(model2.probit, test='Chisq')
model3.probit <- update(model2.probit, .~.-tOut:sex)

drop1(model2.cloglog, test='Chisq')
model3.cloglog <- update(model2.cloglog, .~.-tOut:sex)

### 3. selection
```

```
drop1(model3.logit, test='Chisq')
model4.logit <- update(model3.logit, .~.-tOut)

drop1(model3.probit, test='Chisq')
model4.probit <- update(model3.probit, .~.-tOut)

drop1(model3.cloglog, test='Chisq')
model4.cloglog <- update(model3.cloglog, .~.-tOut)

### 4. selection

drop1(model4.logit, test='Chisq')
model5.logit <- update(model4.logit, .~.-time)

drop1(model4.probit, test='Chisq')
model5.probit <- update(model4.probit, .~.-time)

drop1(model4.cloglog, test='Chisq')
model5.cloglog <- update(model4.cloglog, .~.-time)

### 4. selection

drop1(model5.logit, test='Chisq')
model6.logit <- update(model5.logit, .~.-sex:tInOp)

drop1(model5.probit, test='Chisq')
model6.probit <- update(model5.probit, .~.-sex:tInOp)

drop1(model5.cloglog, test='Chisq')
model6.cloglog <- update(model5.cloglog, .~.-sex:tInOp)

### 5. selection

drop1(model6.logit, test='Chisq')
model7.logit <- update(model6.logit, .~.-tInOp)

drop1(model6.probit, test='Chisq')
model7.probit <- update(model6.probit, .~.-tInOp)

drop1(model6.cloglog, test='Chisq')
model7.cloglog <- update(model6.cloglog, .~.-tInOp)

### 6. SUMMARY

summary(model7.logit)
summary(model7.probit)
summary(model7.cloglog)
```

```
### PLOT OF BEST ####

png("figures/res_binom_2.png", height = 1000, width = 1200, pointsize =
    ↪ 20)
layout(matrix(c(1,2,1,3),2,2))
plot(residuals(model7.logit) ~ ozone$Temp, xlab = "Temp", ylab = "
    ↪ Residual")
lines(seq(min(ozone$Temp), max(ozone$Temp), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$Temp, span = 0.5, degree = 1),
          ↪  seq(min(ozone$Temp), max(ozone$Temp), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
plot(residuals(fit1) ~ ozone$Hum, xlab = "Hum", ylab = "Residual")
lines(seq(min(ozone$Hum), max(ozone$Hum), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$Hum, span = 0.5, degree = 1),
          ↪ seq(min(ozone$Hum), max(ozone$Hum), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
plot(residuals(fit1) ~ ozone$InvHt, xlab = "InvHt", ylab = "Residual")
lines(seq(min(ozone$InvHt), max(ozone$InvHt), length.out = 100),
      predict(loess(residuals(fit1) ~ ozone$InvHt, span = 0.5, degree = 1)
          ↪ , seq(min(ozone$InvHt), max(ozone$InvHt), length.out = 100)),
      col = "red", lwd = 0.5)
grid()
dev.off()

################## POISSON ###################

setwd("/Users/andreasengly/Documents/Danmarks␣Tekniske␣Universitet/2.␣
    ↪ semester/02424␣Videregende␣dataanalyse/Assignments/Assignment␣2")
data <- read.csv('./dat_count.csv', sep = ";")

#### LINK: Log

model1 <- glm(clo ~ time + nobs + tOut*sex + tInOp*sex, data = data,
    ↪ family = poisson(link = "log"))
summary(model1)

print(paste("The␣p-value␣is:␣", round(1 - pchisq(model1$deviance, model1$
    ↪ df.residual), 5)))

drop1(model1, test='Chisq')
summary(model1)

model2 <- update(model1, .~.-nobs)
summary(model2)
drop1(model2, test='Chisq')
```

```
model3 <- update(model2, .~.-tOut:sex)
summary(model3)
drop1(model3, test='Chisq')

model4 <- update(model3, .~.-tOut)
summary(model4)
drop1(model4, test='Chisq')

model5 <- update(model4, .~.-time)
summary(model5)
drop1(model5, test='Chisq')

model6 <- update(model5, .~.-sex:tInOp)
summary(model6)
drop1(model6, test='Chisq')

model7 <- update(model6, .~.-tInOp)
summary(model7)
drop1(model7, test='Chisq')

summary(model7)
print(paste("The␣p-value␣is:␣", round(1 - pchisq(model7$deviance, model7$
    ↪ df.residual), 5)))

plot(model8$fitted.values, model8$residuals, xlab="Fitted␣values", ylab="
    ↪ Residuals")

model8$linear.predictors

plot(data$clo)

boxplot(data$clo[data$sex == 'male'], data$clo[data$sex == 'female'],
        names = c("Males", "Females"), ylab = '#␣changes␣of␣clothes',
        col = c('salmon', 'salmon'))
grid()

sd(data$clo[data$sex == 'male'])
sd(data$clo[data$sex == 'female'])

#### LINK: Log
#### Offset: time

model1 <- glm(clo ~ offset(time) + nobs + tOut*sex + tInOp*sex, data =
    ↪ data, family = poisson(link = "log"))
summary(model1)

print(paste("The␣p-value␣is:␣", round(1 - pchisq(model1$deviance, model1$
    ↪ df.residual), 5)))
```

```
drop1(model1, test='Chisq')
summary(model1)

model2 <- update(model1, .~.-nobs)
summary(model2)
drop1(model2, test='Chisq')

model3 <- update(model2, .~.-tOut:sex)
summary(model3)
drop1(model3, test='Chisq')

model4 <- update(model3, .~.-tOut)
summary(model4)
drop1(model4, test='Chisq')

model5 <- update(model4, .~.-sex:tInOp)
summary(model5)
drop1(model5, test='Chisq')

model6 <- update(model5, .~.-tInOp)
summary(model6)
drop1(model6, test='Chisq')

summary(model6)
print(paste("The␣p-value␣is:␣", round(1 - pchisq(model6$deviance, model6$
    ↪ df.residual), 5)))

#### Par 3 ----
fan.dat = read.csv("CeilingFan.csv", sep = ";")

## 3.1
fan.dat$TSV = factor(paste0(fan.dat$TSV,".tsv"))
fan.dat$fanSpeed = factor(paste0(fan.dat$fanSpeed,".fanSpeed"))

con.tab = table(fan.dat$fanSpeed, fan.dat$TSV)
addmargins(con.tab)

#COLUMN PERCENTAGES
colpercent<-prop.table(con.tab, 2)
colpercent
row.names(colpercent) = 0:2

png("figures/con-tab3.png", height = 1000, width = 1000, pointsize = 24)
barplot(t(colpercent), beside = TRUE, col = 2:4, las = 1,
      ylab = "Percent", xlab = "fanSpeed")
legend( legend = paste("TSV␣=",0:2), fill = 2:4,"topright", cex = 1)
dev.off()
```

```
chi <- chisq.test(con.tab, correct = FALSE)
chi


## 3.2
library(ordinal)
names(fan.dat)
m1.p <- clm( TSV ~ 1, data = fan.dat )
summary(m1.p)
fan.dat$fanSpeed = as.numeric(fan.dat$fanSpeed)
m2.p <- clm( TSV ~ fanSpeed, data = fan.dat, link = "logit" )
summary(m2.p)

anova(m1.p,m2.p)


## 3.3
fan.dat = read.csv("CeilingFan.csv", sep = ";")
fan.dat$fanSpeed = (fan.dat$fanSpeed)
fan.dat$fanType = as.factor(fan.dat$fanType)
fan.dat$TSV = as.factor(fan.dat$TSV)
m3.p <- clm( TSV ~ fanSpeed:fanType + fanType, data = fan.dat)
m4.p <- clm( TSV ~ fanSpeed:fanType + fanType, data = fan.dat, link = "
   ↪ probit")
m5.p <- clm( TSV ~ fanSpeed:fanType + fanType, data = fan.dat, link = "
   ↪ cloglog")
AIC(m3.p)
AIC(m4.p)
AIC(m5.p)
drop1(m3.p, test = "Chisq")
drop1(m4.p, test = "Chisq")
drop1(m5.p, test = "Chisq")

summary(m5.p)


## 3.4
ex.n = 10
x.new = data.frame(fanSpeed = rep(0:(ex.n-1),2),
                   fanType = c(rep("downstream",ex.n),rep("upstream",ex.n))
                      ↪ )
x.new$fanSpeed = as.numeric(x.new$fanSpeed)
x.new$fanType = as.factor(x.new$fanType)
pred.new = predict(m4.p, newdata = x.new)

col.1 = "blue"
col.2 = "red"
```

```
png("figures/clm3.png", height = 800, width = 1000, pointsize = 24)
plot(0:(ex.n-1),pred.new$fit[1:ex.n,3], type = 'l',
     ylab = "Probability", xlab = "fanSpeed", ylim = c(0,1), lty = 6, col
        ↪ = col.1)
points(0:(ex.n-1),pred.new$fit[1:ex.n,3], col = col.1)
lines(0:(ex.n-1),pred.new$fit[1:ex.n,2], lty = 2, col = col.1)
points(0:(ex.n-1),pred.new$fit[1:ex.n,2], lty = 2, col = col.1, pch = 2)
lines(0:(ex.n-1),pred.new$fit[1:ex.n,1], lty = 3, col = col.1)
points(0:(ex.n-1),pred.new$fit[1:ex.n,1], col = col.1, pch = 3)

lines(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),3], col = col.2, lty = 6)
points(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),3], col = col.2, pch =
    ↪ 1)
lines(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),2], lty = 2, col = col.2)
points(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),2], lty = 2, col = col
    ↪ .2, pch = 2)
lines(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),1], lty = 3, col = col.2)
points(0:(ex.n-1),pred.new$fit[(ex.n+1):(2*ex.n),1], col = col.2, pch =
    ↪ 3)
grid()
legend("topright", c("Downstream", "Upstream","TSV=0","TSV=1","TSV=2"),
    ↪ pch=c(-1,-1,3,2,1),
       lty = c(1,1,3,2,6), col = c("blue","red",1,1,1), bg = "white")
dev.off()
```