BDA Project work: The effect of weather on city bike usage

Matias Ahonen, Georgy Ananov, Ida Granö

Contents

Introduction

Motivation

City bikes have become a popular way of traveling in the Helsinki region over the past few years. Currently there are around 4600 bikes and 460 bike stations in use all over the city (HSL, 2021). The city bikes provide an alternative to conventional public transportation, offering a quick and easy way to move within the city. The system is simple: it is possible to rent share-use bikes by buying a pass for a whole season, for a week, or for a single day, for up to 30 minutes per use, or up to five hours for an additional charge. City bikes are available from April until end of October each year, and they were first introduced in year 2016. The number of bikes and bike stations has been steadily increasing over the last 5 years.

Unlike buses and trains, city bikes are not restricted to specific routes or schedules. This presents a number of problems for urban planners, traffic controllers and public transport engineers. The demand for city bikes varies from day to day, which necessitates that the biking infrastructure is built to function in an optimal way under drastically differing load conditions. Unexpected spikes or dips in the bike usage could also affect the operation of other public transport systems. The need to be able to predict city bike demand then is quite apparent.

Problem description

The main goal of the project explore the biking data and to come up with a method for forecasting the number of city bike trips that occur in one day.

Main modeling idea

One factor that is very likely to affect how willing an individual is to choose a city bike as a mode of transportation for the day is the current weather conditions. People are more likely to opt for biking when the weather is favorable, while cold or rainy weather would push people to use methods of transportation that shelter them from the elements (See Figures 1-4).

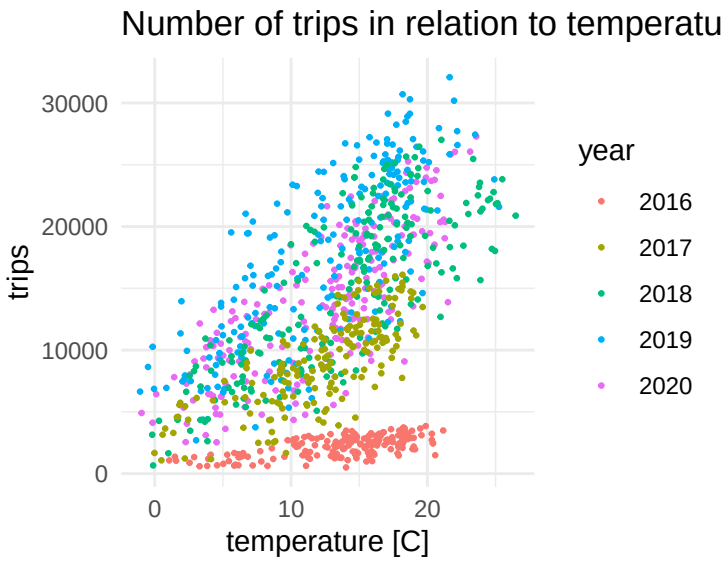

Figure 1: The number of daily bike trips against temperature in years 2016-2020.

In this project, we are interested in modeling how weather parameters, such as temperature, precipitation and humidity, affects the use of city bikes.

Data description

We retrieved data on city bike usage from kaggle.com, and data on daily weather from ilmatieteenlaitos.fi. The biking dataset includes information on every bike trip since the introduction of the city bikes in 2016 until late 2020, with 14 variables (departure time, return time, departure station, return station, distance traveled, duration, average speed, departure latitude, departure longitude, return latitude, return longitude, and air temperature). We pulled three variables form *ilmatieteenlaitos.fi*: daily mean temperature, daily precipitation, and mean daily humidity.

We were interested in modeling the daily use of city bikes, and therefore pre-processed the biking dataset to get daily values. We calculated the total number of trips each day based on the date in the departure time to get the total number of trips per day, and for each day, we averaged the duration, speed and distance traveled. Because the precipitation data was heavily focused on small values, we applied a log transform to even out the distribution, after adding a small value (0.1) to avoid the log of 0. We then combined this data with the daily weather variables.

```
head(combined_trunc[c("date", "n_trips", "Precipitation", "MeanTemp", "Humidity")])
```

```
##         date n_trips Precipitation MeanTemp Humidity
## 1 2020-03-23    2552     -2.302585 2.270833 69.87500
## 2 2020-03-24    3115     -2.302585 3.745833 77.16667
## 3 2020-03-25    3625     -2.302585 5.183333 69.54167
## 4 2020-03-26    4782     -2.302585 4.791667 75.62500
## 5 2020-03-27    4826     -2.302585 4.495833 69.20833
## 6 2020-03-28    7010     -2.302585 5.379167 58.70833
```

Figures 1, 2, 3 and 4 illustrate the data. As can be seen in Figure 1 and 2, year 2016, when the bikes were first introduced, is an outlier in terms of bike trip numbers, and was therefore excluded from analysis. Figure 2 shows the bike usage over time, Figure 3 shows the number of trips against the daily temperature and precipitation, and Figure 4 shows the number of daily trips against the average daily humidity.

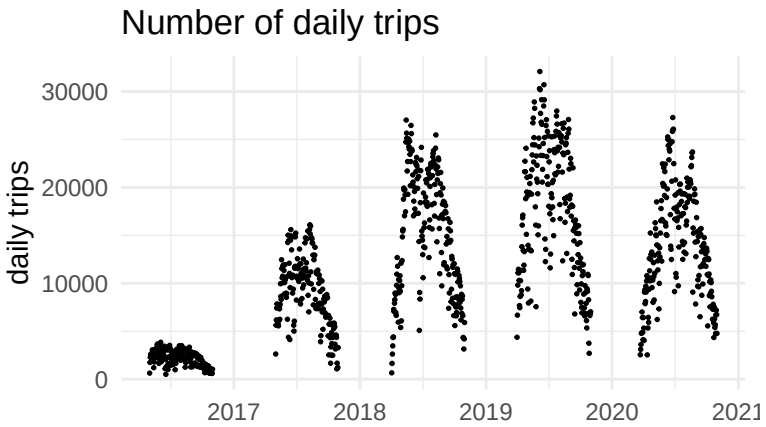

Figure 2: The daily city bike usage in number of trips over time, from 2016 to end of 2020. The bikes are unavailable during the winter months (November until end of March). The usage of city bikes has clearly grown the first few years.

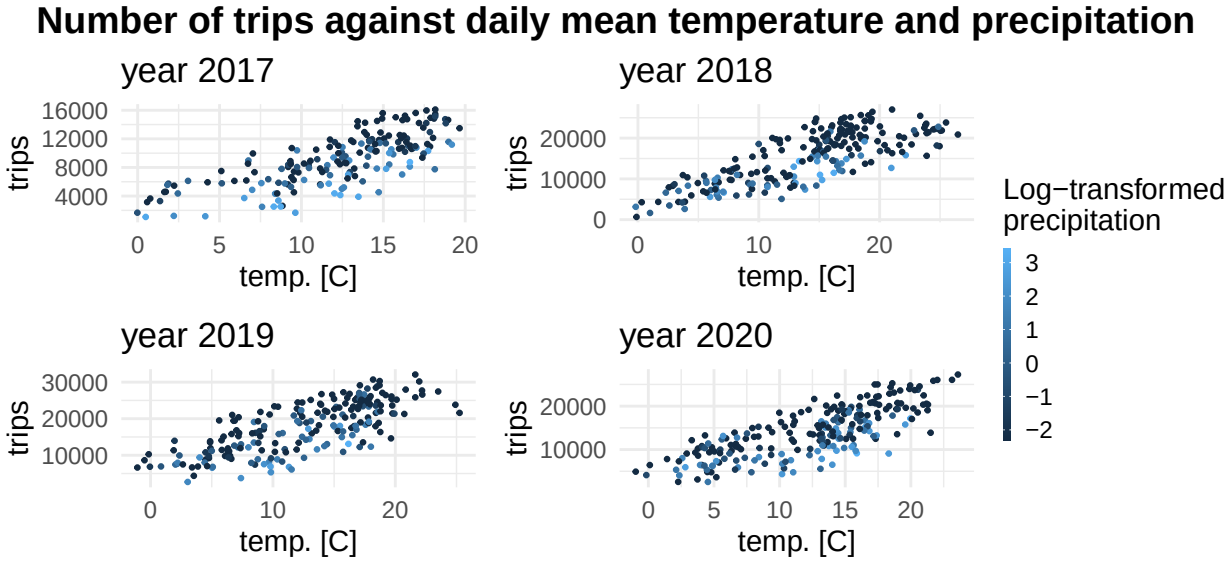

Figure 3: The number of biking trips plotted against temperature over years 2017 to 2020. The color indicates the logarithm of level of precipitation, after adding a small value to avoid the logarithm of 0 (ln(precipitation+0.1)). The relationship between temperature and biking trips seems to be rouhgly linear. It seems like there are less biking trips on rainy days.

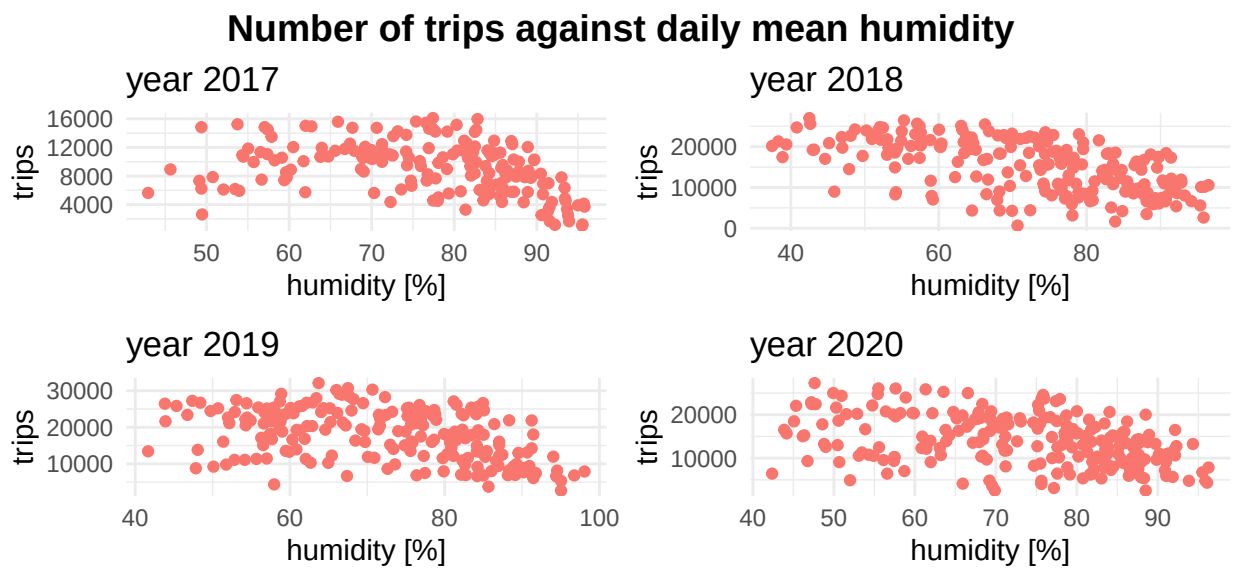

Figure 4: The number of biking trips plotted against humidity over years 2017 to 2020. The number of bike trips seems to go down with humidity.

The biking dataset has previously been used in descriptive analyses of biking behaviour (e.g., mean distance and duration, bike usage in different months, weekdays, and times of day), for analysis of popular destinations, and for network analysis. To our knowledge, a deeper exploration of the effects of weather on city bike usage has not been done before.

Model introduction

In this project, we are comparing three models of increasing complexity. First, we build a simple pooled univariate model with temperature as the predictor of the number of trips. Then, we add humidity and precipitation to the linear pooled model as covariates. Finally, as the number of bikes and bike stations have grown over the years, we build a hierarchical model for investigating the data. Below is the mathematical notations for all three models.

Model 1 Model 2: Model 3:

Hyper-priors:

$$\mu_\alpha \sim N(1840, 1118.54)$$
$$\sigma_\alpha \sim HalfCauchy(0, 1000)$$
$$\mu_\beta \sim N(-1400, 851.06)$$
$$\sigma_\beta \sim HalfCauchy(0, 1000)$$
$$\mu_\gamma \sim N(-1000, 500)$$
$$\sigma_\gamma \sim HalfCauchy(0, 1000)$$
$$\mu_\delta \sim N(0, 10000)$$
$$\sigma_\delta \sim HalfCauchy(0, 1000)$$

Priors:

$\alpha \sim N(1840, 1118.54)$	$\alpha \sim N(1840, 1118.54)$	$\alpha \sim N(\alpha_0, \sigma_0)$
	$\beta \sim N(-1400, 851, 06)$	$\beta \sim N(\beta_0, \sigma_0)$
	$\gamma \sim N(-1750, 1063.83)$	$\gamma \sim N(\gamma_0, \sigma_0)$
$\delta \sim N(0, 10000)$	$\delta \sim N(0, 10000)$	$\delta \sim N(\delta_0, \sigma_0)$
$\sigma \sim HalfCauchy(0, 1000)$	$\sigma \sim HalfCauchy(0, 1000)$	$\sigma \sim HalfCauchy(0, 1000)$

Linear model and likelihood:

$\mu = \alpha * Temp + \delta$	$\mu = \alpha * Temp + \beta * \log(Prec+0.1)+\gamma*Hum+\delta$	$\mu = \alpha*Temp+\beta*log(Prec+0.1) + \gamma * Hum + \delta$
$y \sim N(\mu, \sigma)$	$y \sim N(\mu, \sigma)$	$y \sim N(\mu, \sigma)$

Below we describe the reasoning behind our choice of priors:

We estimate priors for the parameters using common sense and prior research.

According to HSL, during 2020 there were almost 3500 bikes in Helsinki and Espoo and each bike was used five times per day in Helsinki and twp times per day in Espoo (HSL, 2020). As most of the bikes are in Helsinki, we estimate that the average number of trips for a bike per day is 4. This would result in the average number of total trips being 4 * 3500 = 14 000.

- α : We believe that a one-degree increase in temperature is unlikely to decrease the number of trips or to increase the number of trips by more than 20% (2800). We estimate the prior distribution of *alpha* by estimating the prior probability for *alpha* to be $\Pr(0 < \alpha < 2800) = 0.9$. This way we get the distribution of Normal(1840, 1118.54) or our prior.
- β : It is common sense that precipitation, decreases the amount of outdoor activity so we believe that is highly unlikely that it would increase the number of trips. We also believe that a single percentage increase in rain is unlikely to decrease the number of trips by over 20%. Thus, we estimate the prior distribution of β by calculating the prior probability so that: $\Pr(-2800 < \beta < 0) = 0.95$. The resulting prior is Normal(-1400, 851.06).
- γ : For humidity it is difficult to estimate how much a single percentage increase in relative humidity, would affect the number of trips made. Based on previous research (see f.e. Flynn, Dana„Sears, & Aultman-Hall, (2012)) we believe that the effect humidity has on number of trips is very likely to be negative. We estimate that a single percentage point increase in humidity most likely decreases the number of trips but doesn't do it by more than 25%. We estimate the prior distribution of γ by estimating the prior probability for $\gamma$to be: $\Pr(-3500 < \gamma < 0) = 0.9$ and get the distribution of Normal(-1750,1063.83).
- δ : For the intercept term δ we don't have prior knowledge, so we decided to go with a very weak prior of Normal(0,10 000). This way we make sure we don't limit the how the intercept is learnt from the data.

For the variance of the daily trips, we also have no prior knowledge, so we use a very weak prior of Half-Cauchy(0,1000).

In the hierarchical model we use the priors estimated for parameters of the pooled model as the hyperpriors of the mean values of the parameters. For the standard deviations we estimate individual variances for each of the parameters that follow the Half-Cauchy(0,1000) distribution. We model the different years with common predictive variance, which follows a Half-Cauchy distribution(0,500)

In the following section, we present and assess the three models in order.

Singe-variable pooled model

Model description

Let us start with a very basic pooled linear model that predicts the daily number of trips based solely on temperature. We run the model with 4 chains, with 2000 posterior draws, out of which 1000 are used as warmup.

Below is the stan code for the model.

```
data {
  int<lower=0> N;      // Number of observations
  real y[N];           // Vector of daily trip numbers
  real t[N];           // Vector of temperature observations
}

parameters {
  real alpha;          // Slope parameter
  real delta;          // Intercept
  real<lower=0> sigma; // Variance
}

transformed parameters {
  real mu[N];
  for (i in 1:N) mu[i] = alpha * t[i] + delta;
}

model {
  // Priors
  alpha ~ normal(1840, 1118.54);
  delta ~ normal(0, 10000);
  sigma ~ cauchy(0, 500);

  // Likelihood
  y ~ normal(mu, sigma);
}

generated quantities {
  vector[N] log_lik;
  for (i in 1:N) {
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
  }
}
```

```
# Preparing the data for use in stan
stan_data <- list(
  N = nrow(combined),
  K = length(unique(combined$group)),
  x = combined$group,
  y = combined$n_trips,
  t = combined$MeanTemp,
  p = combined$Precipitation,
  h = combined$Humidity
)
```

```
# Fitting the model
fit <- stan(file="pooled_T.stan", data=stan_data,
            refresh=0, pars=c("alpha", "delta", "sigma", "log_lik"))
draws <- data.frame(extract(fit))
fit_pooled_T <- fit
draws_pooled_T <- draws
```

Convergence

Let us now take a look at how the four Markov chains behave to ensure that the default number of iterations is enough to reach convergence with this model (see Figure 5).

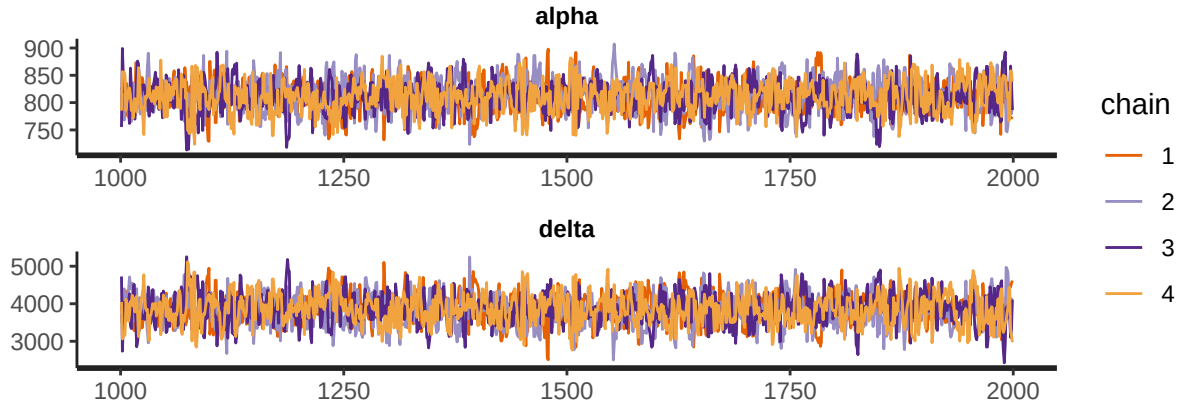

Figure 5: A visualization of the Markov chains for alpha and delta.

Visually the chains appear to have reached convergence. We can further confirm this by examining the $\hat{R}$ diagnostic. The diagnostic compares the within-chain and between-chain estimates of the model parameters. The more the estimates differ, the larger the $\hat{R}$ diagnostic. Generally an $\hat{R}$ value of less than 1.05 is accepted as evidence for convergence.

```
summary_1 <- summary(fit, probs=c(0.05, 0.95), pars=c("alpha", "delta"))
summary_1$summary
```

```
##             mean    se_mean        sd        5%       95%    n_eff     Rhat
## alpha  811.9943  0.7750331  29.36498  763.3228  861.0587 1435.552 1.003568
## delta 3851.2431 10.8528232 407.92265 3176.6419 4522.4377 1412.766 1.003066
```

We can confirm that both of the $\hat{R}$ are below the recommended threshold of 1.05, which indicates full convergence. We can also confirm a sufficiently large effective sample size for both parameters.

Posterior predictive checks

Let's check how the posterior draws of the model parameters line up against the original data (see Figure 6).

```
## No id variables; using all as measure variables
```

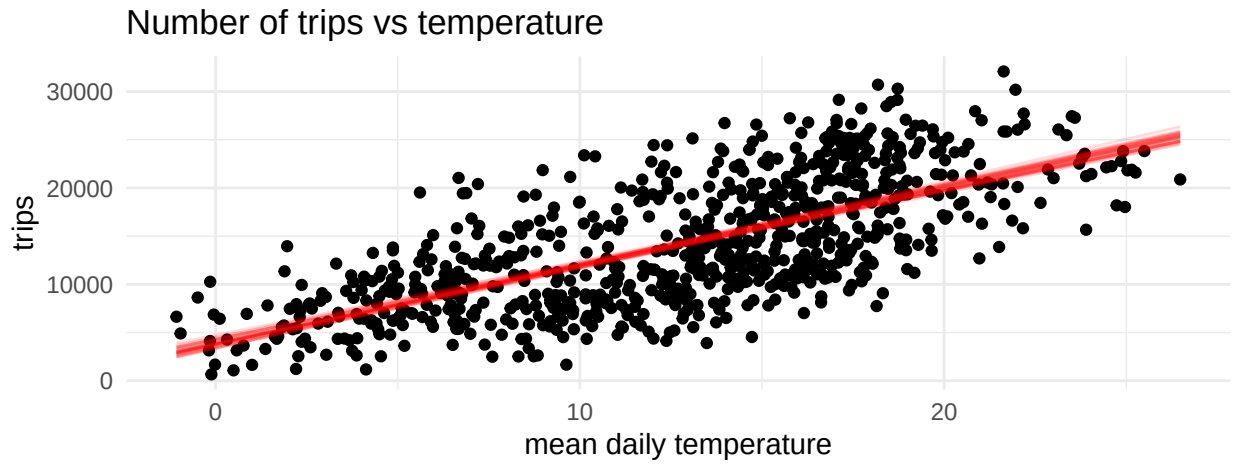

Figure 6: The linear fit of the effect of mean daily temperature on the number of daily bike trips, based on the posterior draws of alpha and delta.

Cross-validation

In order to compare the efficiency of this model to other models created for this task, we can use the Pareto-Smoothed Leave-One-Out Cross Validation method. The Pareto-k values are shown in Figure 7.

```
## PSIS-LOO elpd for the univariate pooled model: -8163.12619623651
```

```
## p_loo for the univariate pooled model model: 2.44564171018357
```

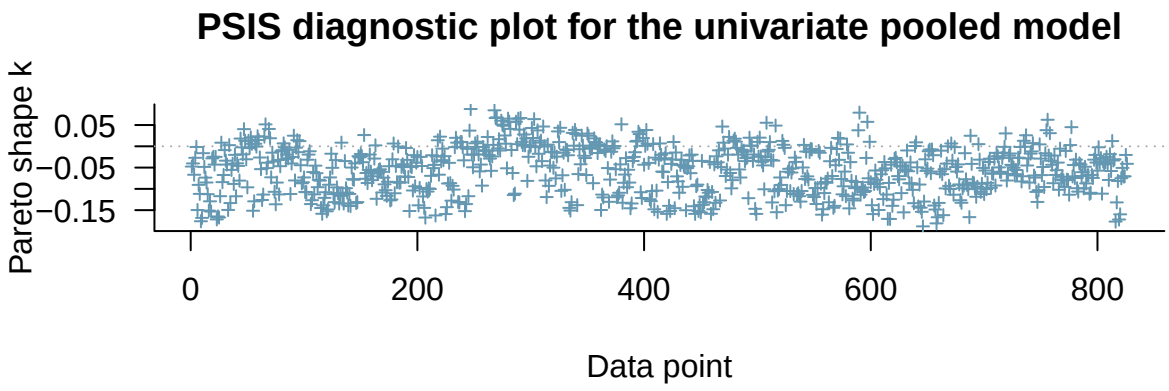

Figure 7: The PSIS-LOO values for the pooled univariate model.

The observed Pareto-k values all fell far below the 0.7 threshold, which indicates that the PSIS-LOO results are reliable. Once we obtain the diagnostics for the other models, we will be able to compare them and discern the most effective model.

Multi-variable pooled model

Model description

As seen in Figures 2 and 3, temperature is not the only variable to affect biking trip numbers. To improve the accuracy of the predictions, we decided to include more covariates into the list of model parameters. We are still using a linear pooled model, but this time we are basing the predictions on precipitation and humidity data in addition to the temperature observations. Again, we run the model with 4 chains, with 2000 posterior draws, out of which 1000 are used as warmup.

The code for the stan model is included below:

```
data {
  int<lower=0> N;      // Number of observations
  real y[N];           // Vector of daily trip numbers
  real t[N];           // Vector of temperature observations
  real p[N];           // Vector of precipitation observations
  real h[N];           // Vector of humidity observations
}

parameters {
  real alpha;          // Slope parameter along the temperature axis
  real beta;           // Slope parameter along the precipitatin axis
  real gamma;          // Slope parameter along the humidity axis
  real delta;          // Intercept
  real<lower=0> sigma; // Variance
}

transformed parameters {
  real mu[N];
  for (i in 1:N) mu[i] = alpha * t[i] + beta * p[i] + gamma * h[i] + delta;
}

model {
  // Priors
  alpha ~ normal(1840, 1118.54);
  beta ~ normal(-1400, 851.06);
  gamma ~ normal(-1750,1063.83);
  delta ~ normal(0, 10000);
  sigma ~ cauchy(0, 500);

  // Likelihood
  y ~ normal(mu, sigma);
}

generated quantities {
  vector[N] log_lik;
  for (i in 1:N) {
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
```

```
  }
}
```

```
fit <- stan(file="pooled_TPH.stan", data=stan_data,
            refresh=0, pars=c("alpha","beta","gamma", "delta", "sigma", "log_lik"))
draws <- data.frame(extract(fit))

fit_pooled_TPH <- fit
draws_pooled_TPH <- draws
```

Convergence

Let us now take a look at how the four Markov chains behave to ensure that the default number of iterations is enough to reach convergence with this model in Figure 8.

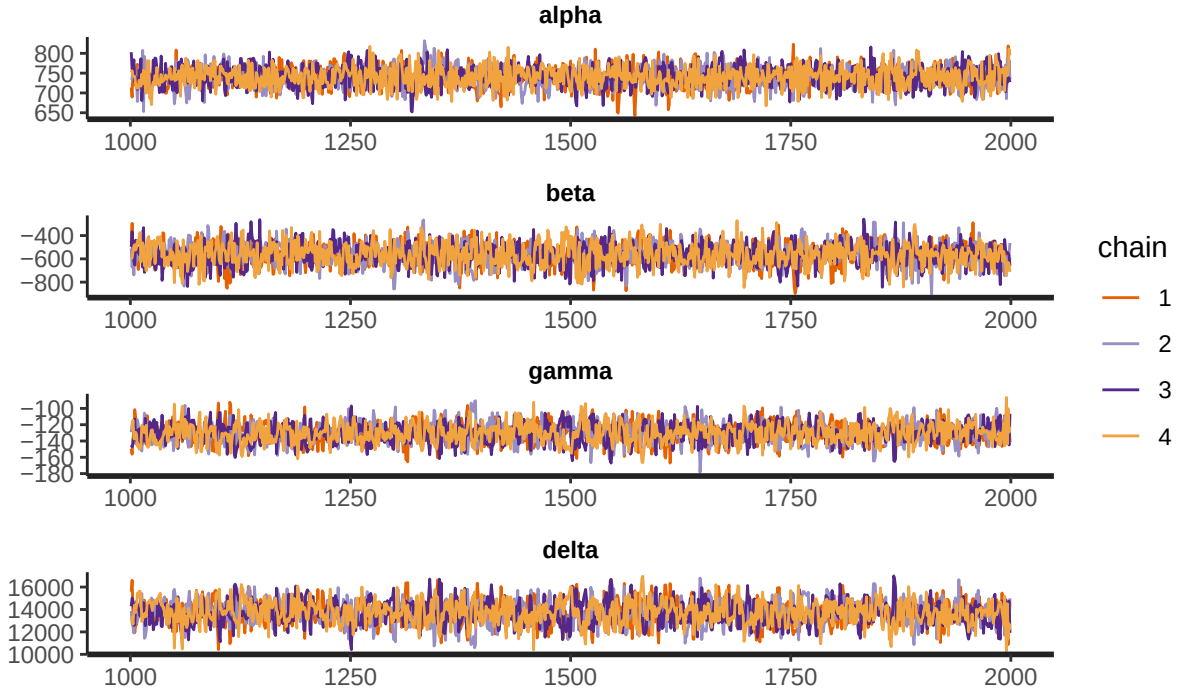

Figure 8: A visualization of the Markov chains for alpha, beta, gamma and delta.

Visually the chains appear to have reached convergence. Let's take a look at the $\hat{R}$ diagnostic to confirm.

```
summary_2 <- summary(fit, probs=c(0.05, 0.95),
                     pars=c("alpha","beta","gamma", "delta"))
summary_2$summary
```

```
##              mean    se_mean         sd         5%        95%    n_eff      Rhat
## alpha    740.6458  0.5011604   25.60842   698.4608   782.2734 2611.030 1.0010653
```

```
## beta   -565.1750  1.9893285   94.37939  -722.5872  -412.0988 2250.823 0.9998765
## gamma  -130.4197  0.2846157   11.96799  -149.7382  -110.3751 1768.173 1.0001774
## delta 13739.7642 25.1438537 1028.50218 12022.5258 15391.9624 1673.196 1.0004688
```

We can observe that all of the $\hat{R}$ values are below the recommended threshold of 1.05, which indicates full convergence. We can also confirm a sufficiently large effective sample size for all parameters.

Posterior predictive checks

Again, let's check how the posterior draws of the model parameters line up against the original data to make sure that the model is sensible in Figures 9-11.

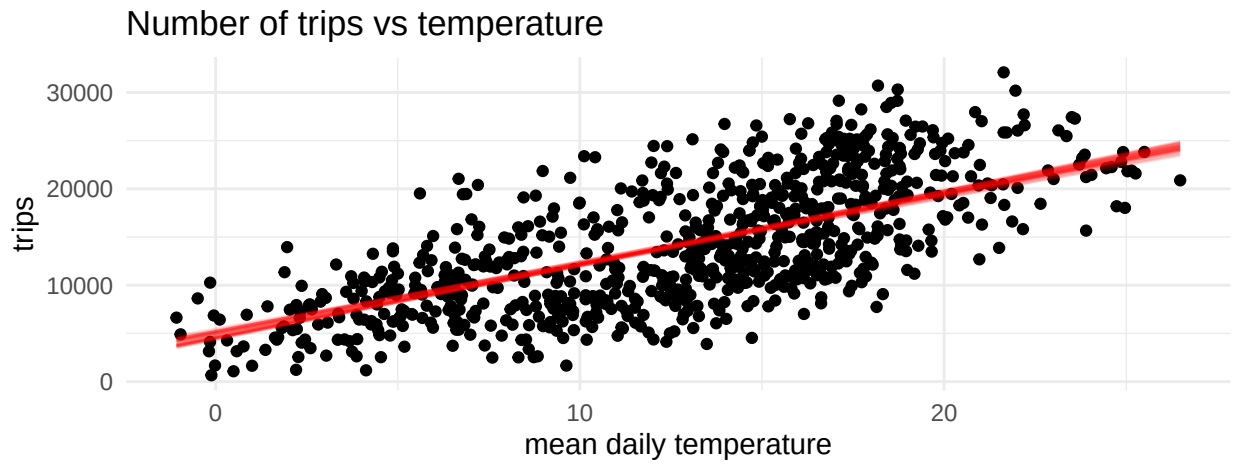

Figure 9: The linear fit of the effect of mean daily temperature on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

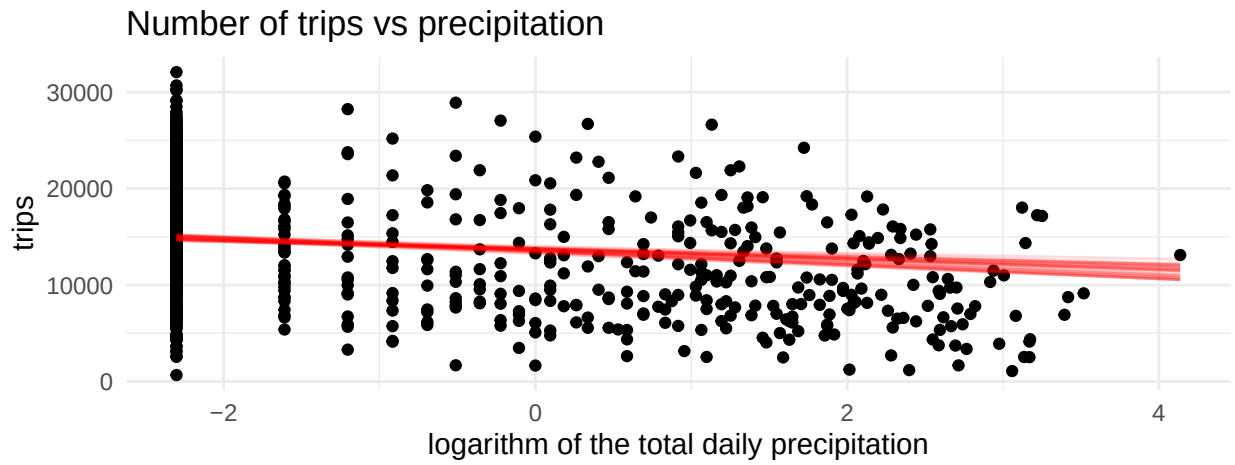

Figure 10: The linear fit of the effect of mean daily precipitation on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

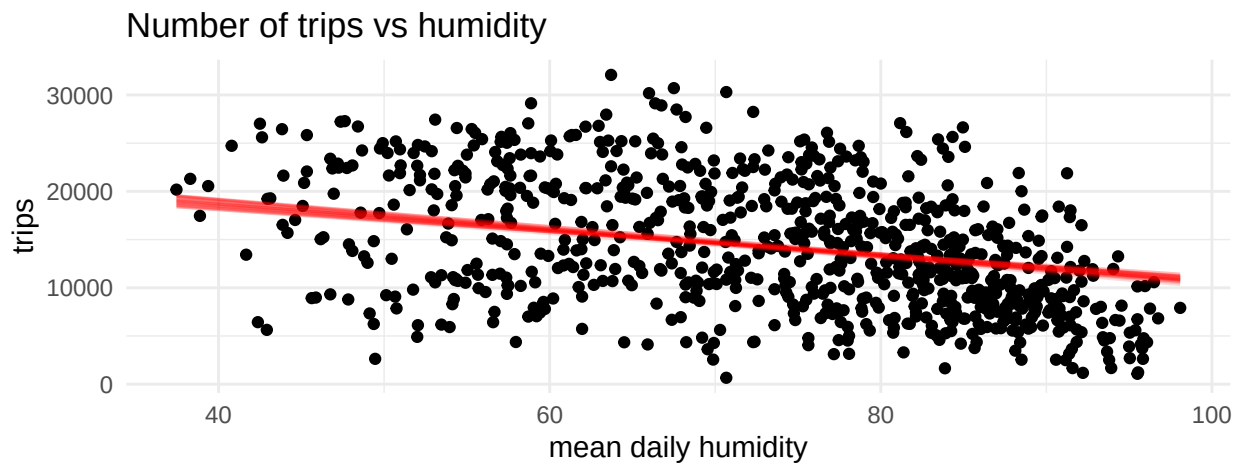

Figure 11: The linear fit of the effect of mean daily humidity on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

Cross-validation

In order to compare the efficiency of this model to other models created for this task, we can use the Pareto-Smoothed Leave-One-Out Cross Validation method. The Pareko-k values are shown in Figure 12.

```
## PSIS-LOO elpd for the multivariate pooled model: -8038.98712464677
```

```
## p_loo for the multivariate pooled model model: 4.48420088357683
```

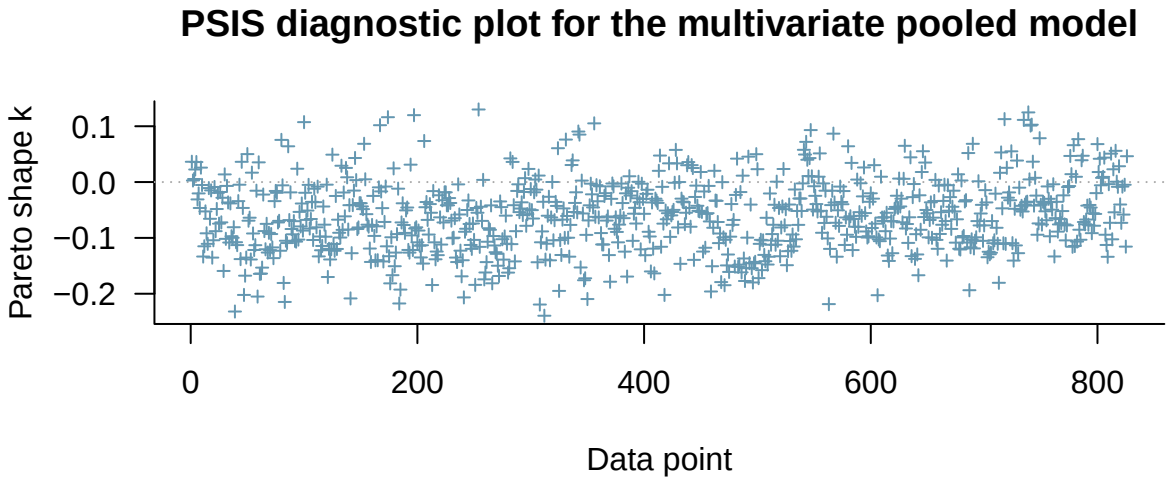

Figure 12: The Pareto-k values for the multivariate pooled model.

The observed Pareto-k values all fell far below the 0.7 threshold, which indicates that the PSIS-LOO results are reliable.

The hierarchical model

Model description

In an attempt to further improve the quality of the predictions, we decided to deploy a hierarchical model, where the data is grouped by years. During the exploration of the data, we noticed that the datapoints in different years follow the same overall pattern, but exhibit slight variations in specific details of the pattern, which is why we felt that using a hierarchical model makes sense. We use all three weather variables to make predictions, just like with the second model.

The code for the hierarchical model is as follows:

```
data {
  int<lower=0> N;            // Total number of observations
  int<lower=0> K;            // Number of years
  int<lower=1,upper=K> x[N]; // Year group indicators
  vector[N] y;               // Vector of daily trip numbers
  vector[N] t;               // Vector of temperature observations
  vector[N] p;               // Vector of humidity observations
  vector[N] h;
}

parameters {
  // Hyper parameters
  real mu_alpha;
  real<lower=0> sigma_alpha;
  real mu_beta;
  real<lower=0> sigma_beta;
  real mu_gamma;
  real<lower=0> sigma_gamma;
  real mu_delta;
  real<lower=0> sigma_delta;

  // Model parameters
  vector[K] alpha;         // Slope along the temperature axis
  vector[K] beta;          // Slope along the precipitation axis
  vector[K] gamma;         // Slope along the humidity axis
  vector[K] delta;         // Intercept
  real<lower=0> sigma;     // Common predictive variance for all years
}

transformed parameters {
  vector[N] mu;
  for (i in 1:N)
    mu[i] = alpha[x[i]] * t[i] +
            beta[x[i]] * p[i] +
            gamma[x[i]] * h[i] +
            delta[x[i]];
}

model {
  // Hyper priors
  mu_alpha ~ normal(1840, 1118.54);
  sigma_alpha ~ cauchy(0, 500);
```

```
  mu_beta ~ normal(-1400, 851.06);
  sigma_beta ~ cauchy(0, 500);
  mu_gamma ~ normal(-1000, 500);
  sigma_gamma ~ cauchy(0, 500);
  mu_delta ~ normal(0, 10000);
  sigma_delta ~ cauchy(0, 500);
  sigma ~ cauchy(0, 500);

  // Priors
  alpha ~ normal(mu_alpha, sigma_alpha);
  beta ~ normal(mu_beta, sigma_beta);
  gamma ~ normal(mu_gamma, sigma_gamma);
  delta ~ normal(mu_delta, sigma_delta);

  // Likelihood
  y ~ normal(mu, sigma);
}

generated quantities {
vector[N] log_lik;
  for (i in 1:N)
    log_lik[i] = normal_lpdf(y[i] | mu[i], sigma);
}
```

```
fit <- stan(file="hierarchical_TPH.stan", data=stan_data, iter = 2000,
            warmup = (2000/2), refresh=0)
draws <- data.frame(extract(fit))

fit_hierarchical_TPH <- fit
draws_hierarchical_TPH <- draws
```

Convergence

Let us now take a look at how the four Markov chains behave to ensure that the default number of iterations is enough to reach convergence with this model. Figure 13 shows a visual convergence assessment plot for the year 2017.

```
year2017 <- traceplot(fit,  ncol=1,
                  pars=c("alpha[1]", "beta[1]", "gamma[1]", "delta[1]"))
year2017
```

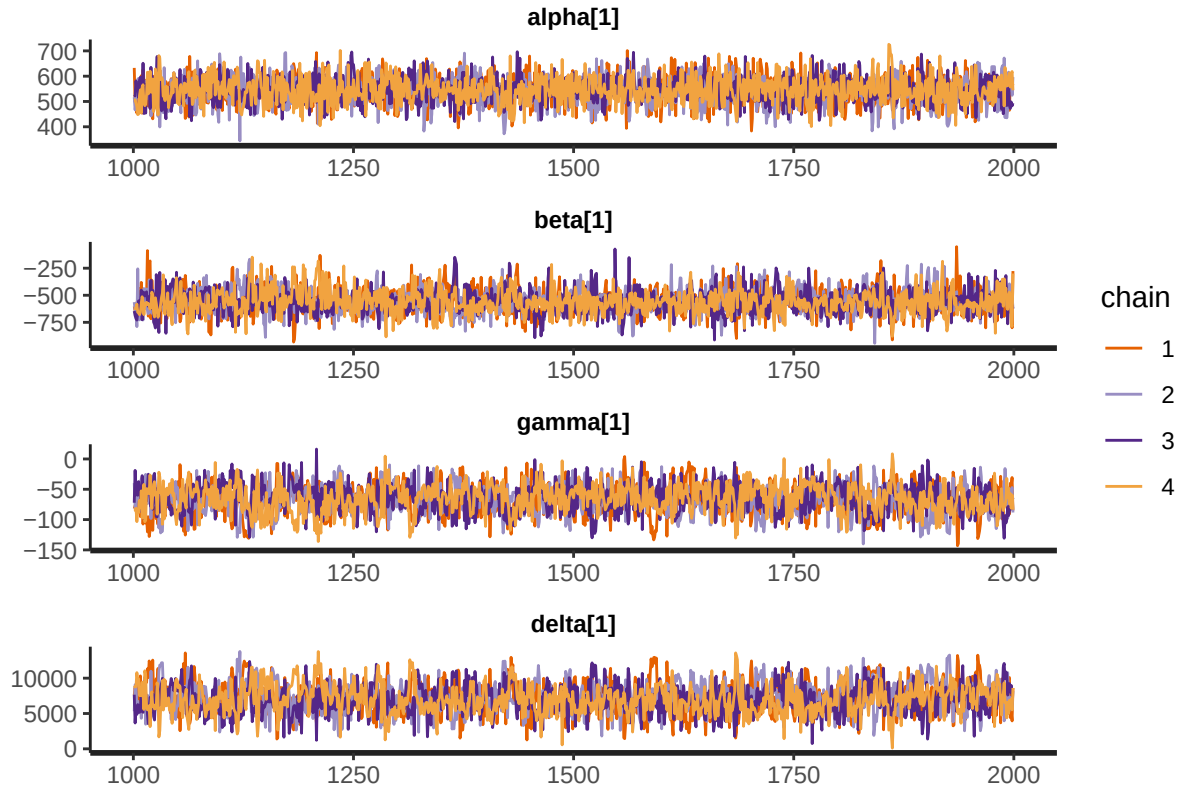

Figure 13: A visualization of the Markov chains for year 2017 for alpha, beta, gamma and delta.

Visually the chains appear to have reached convergence, however, there are parts where the chains seem to behave curiously. We can further check the convergence by examining the $\hat{R}$ diagnostic.

```
summary_3 <- summary(fit, probs=c(),
        pars=c("alpha","beta","gamma", "delta"))
summary_3$summary
```

```
##                mean    se_mean         sd     n_eff      Rhat
## alpha[1]   543.65386  0.9941166   53.54340 2900.9300 1.0009270
## alpha[2]   693.93896  0.5555167   34.31588 3815.8906 0.9997428
## alpha[3]   844.55349  0.6276946   36.30192 3344.7409 1.0002666
## alpha[4]   792.68911  0.5880960   36.12249 3772.7588 0.9991762
## beta[1]   -555.45991  2.7127721  113.10361 1738.3062 1.0013083
## beta[2]   -577.04026  2.1931157  107.66347 2409.9814 0.9998723
## beta[3]   -656.06281  3.4948421  116.88297 1118.5294 1.0023883
## beta[4]   -557.40091  2.0611110  102.44070 2470.2578 0.9999293
## gamma[1]   -66.31538  0.8122625   23.98563  871.9858 1.0028407
## gamma[2]  -122.21826  0.3558682   15.08372 1796.5475 1.0006541
## gamma[3]  -106.99847  0.4394191   16.28630 1373.6846 1.0011591
## gamma[4]  -127.48891  0.3294793   15.41819 2189.8312 0.9995188
## delta[1]  7100.63282 73.3288666 2118.50562  834.6594 1.0035826
## delta[2] 13623.47506 31.6611738 1330.11863 1764.9269 1.0009096
```

```
## delta[3] 14166.04170 38.4745295 1374.90158 1277.0167 1.0013017
## delta[4] 12955.16652 25.7872007 1219.22138 2235.4077 0.9999412
```

All of the $\hat{R}$ are well below the recommended threshold of 1.05, indicating that the iterations have reached convergence. We can also confirm a sufficiently large effective sample size for all parameters, although for a few parameters, the effective sample size seems smaller than for the pooled models.

Posterior predictive checks

Once again, let's check how the posterior draws of the model parameters line up against the original data to make sure that the model is sensible in Figures 14-16.

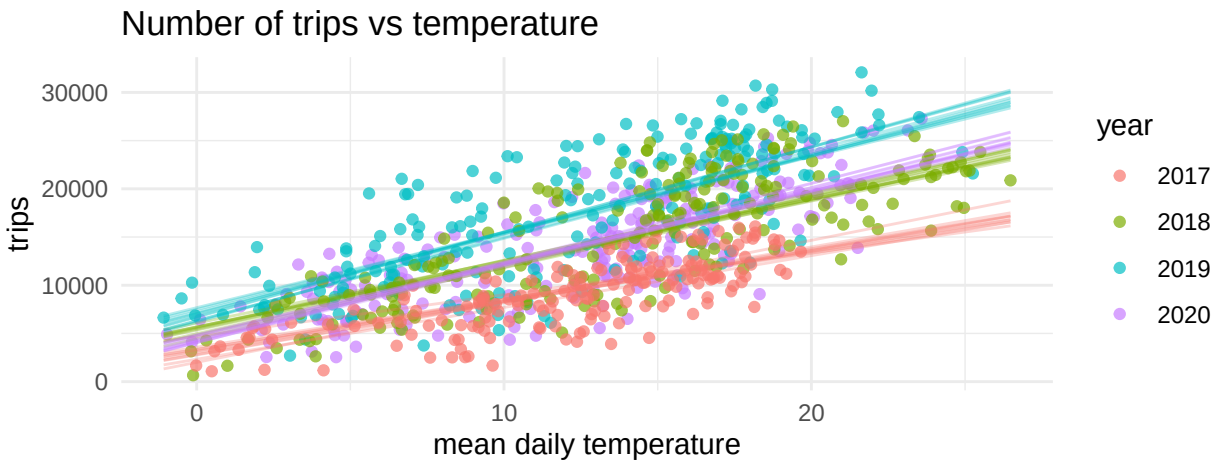

Figure 14: The linear fit of the effect of mean daily temperature on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

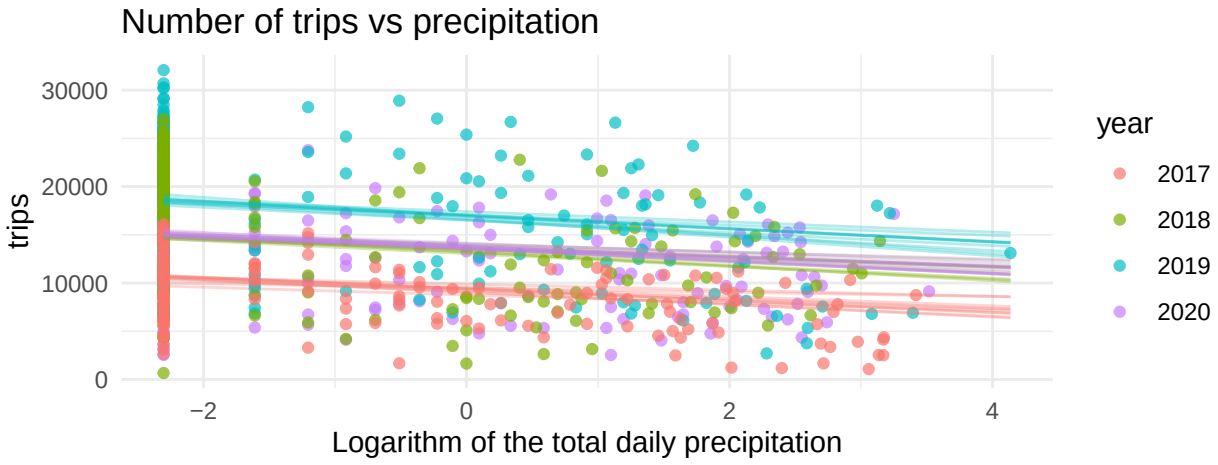

Figure 15: The linear fit of the effect of mean daily precipitation on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

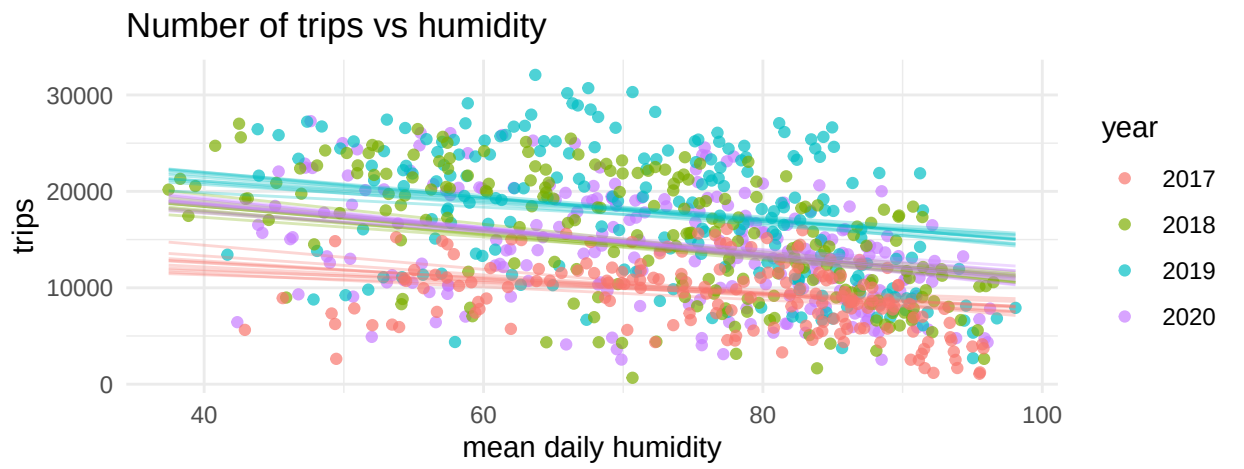

Figure 16: The linear fit of the effect of mean daily humidity on the number of daily bike trips, based on the posterior draws of alpha, beta, gamma and delta.

Cross-validation

Let's take a look at the LOO diagnostics.

```
## PSIS-LOO elpd for the multivariate hierarchical model: -7795.1131741576
```

```
## p_loo for the multivariate hierarchical model model: 14.6107495413705
```

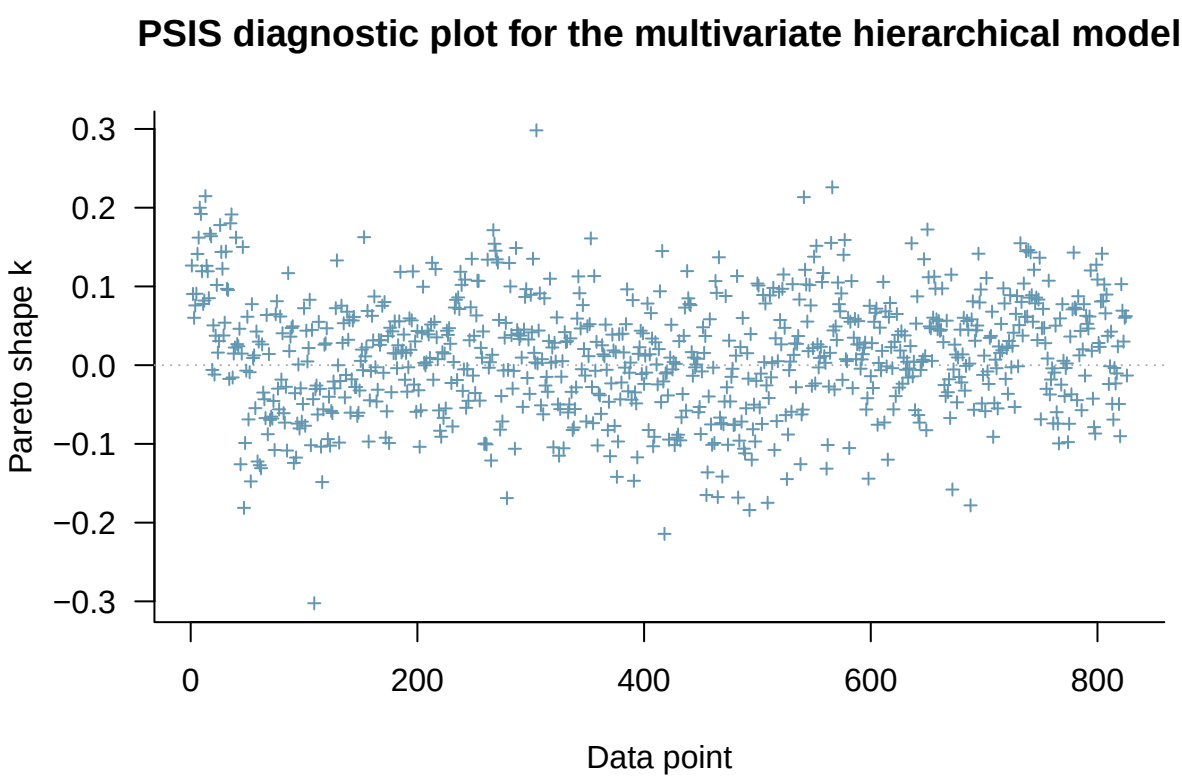

Figure 17: Pareto-k values for the hierarchical model.

All Pareto-k values are below the 0.7 threshold.

Model comparison

Now we can compare the three models. Let's do this by taking a look at the LOO-ELPD values of each model:

```
loo_compare(list(loo_1,loo_2,loo_3))
```

```
##        elpd_diff se_diff
## model3    0.0       0.0
## model2 -243.9      20.4
## model1 -368.0      22.6
```

As expected, we can see that the hierarchical model performs the best, the pooled model with three variables performs the second best and the simple univariate pooled model is the worst model of the three.

Predictive performance assessment

Visual assessment

One way to visually assess the accuracy of the model performance is to use the models to "predict" the daily trip numbers based on the already observed weather conditions. We can then compare the shapes of the predicted distributions to the actual observed trip numbers that were plotted in Figure 1.

First, let's take a look at how the simple pooled model behaves.

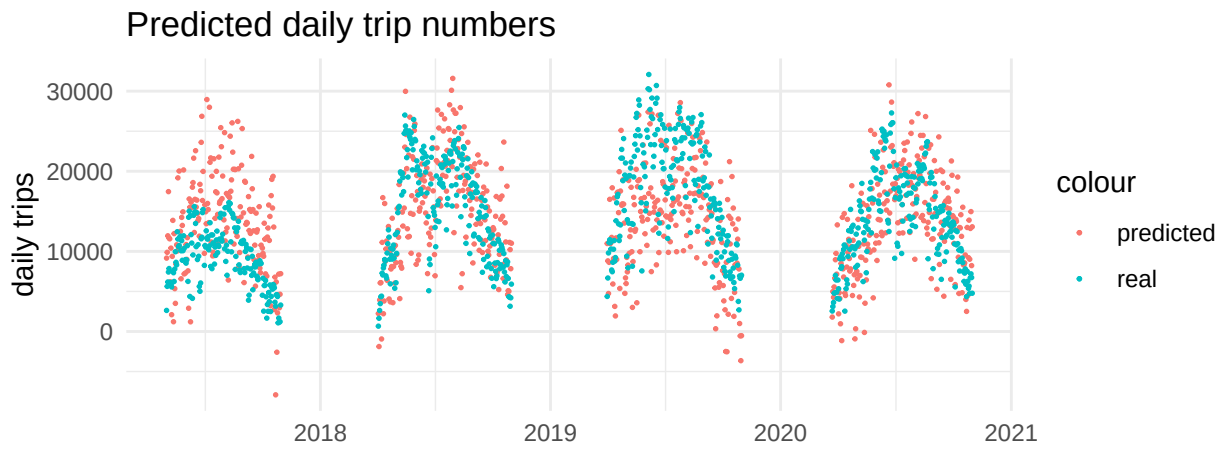

Figure 18: Comparison between predicted number of trips of the univariate pooled model and the actual number of trips. The predicted numbers follow the same shape as the real ones. However, the precited values sometimes fall below zero, and do not distinguish between the yearly trends.

As we can observe, the distributions shapes for years 2017-2020 roughly match the observed data, which indicates that the model is sensible. However, the between-year differences are lost with this model, which causes particularly noticeable difference between predictions and observations during year 2017.

Next, we can go through with similar analysis for the advanced pooled model.

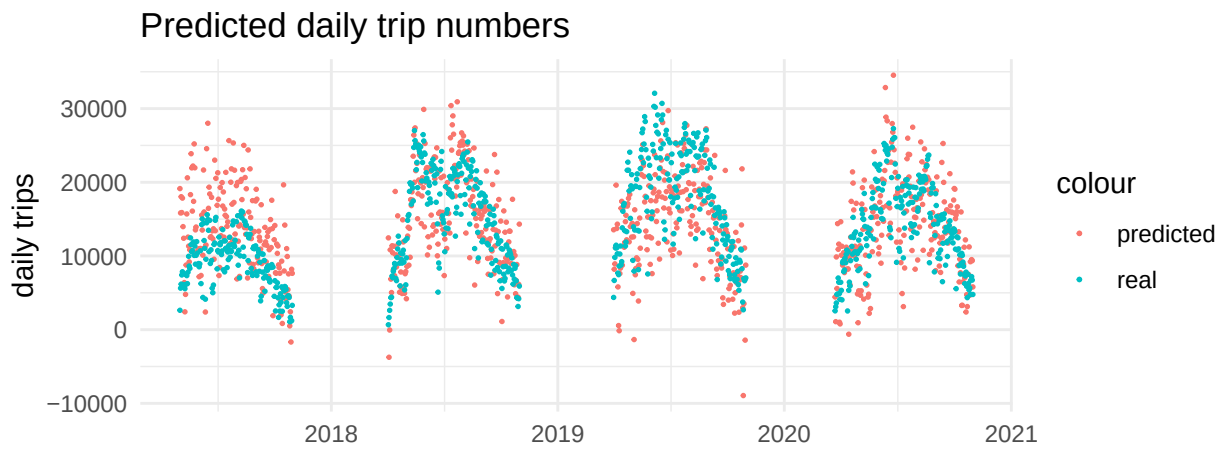

Figure 19: Comparison between predicted number of trips of the multivariate pooled model and the actual number of trips. The predicted numbers follow the same shape as the real ones. However, the precited values do not distinguish between the yearly trends.

As we can observe, the distributions shapes for years 2017-2020 roughly match the observed data, which indicates that the model is sensible. However, the between-year differences are still not accounted for.

Finally, let us take a look at the hierarchical model.

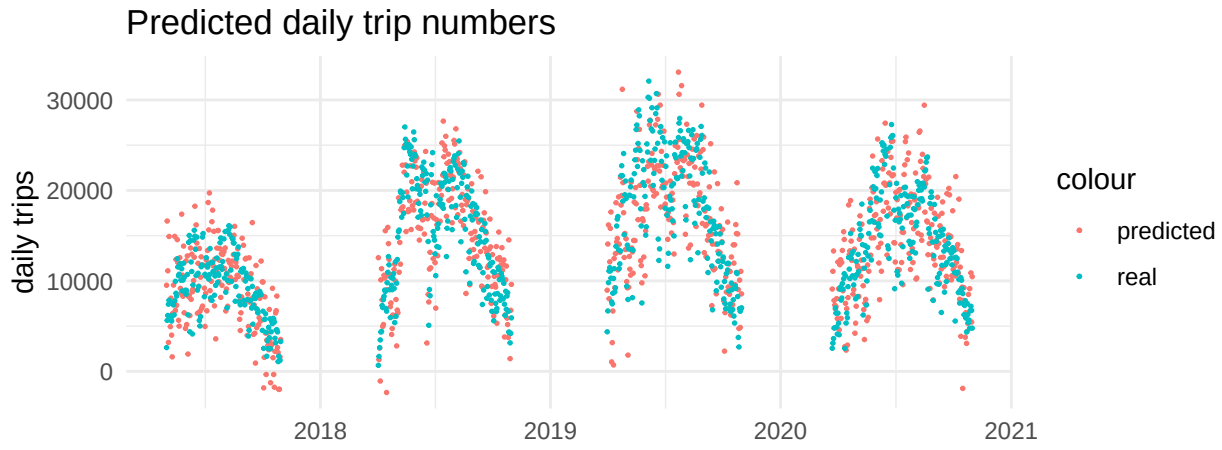

Figure 20: Comparison between predicted number of trips of the hierarchical model and the actual number of trips.

We can now observe that the model effectively accounts for the between-years differences in city bike usage.

Mean Absolute Error

To obtain a more concrete numerical evaluation of the model performance, we can calculate the Mean Absolute Error of the models. Let us take a look at the results:

```
## Mean Absolute Error for the hierarchical model: 3400.39496466223
```

Given that most days see more than 10,000 trips, the observed error should be acceptable in the context of predicting high-demand and low-demand days.

R Squared Diagnostic

We can also evaluate how well our model explains the variance within the data. For that, we will turn to the adjusted R-squared diagnostic.

```
## Adjusted R-squared diagnostic for the hierarchical model: 0.79239505586471
```

We can conclude that majority of observations fit the constructed model well.

Sensitivity analysis

Our priors were based on rough estimations and different choices could easily be argued. To check whether or not our prior choices affect the results significantly, let's test some different priors. Let's do this sensitivity analysis with the hierarchical model, as it was proven to be the best performing one.

Now instead of using the priors we estimated, let's use very weakly informative hyperpriors of μ = Normal(0,10000) and σ = Half-Cauchy(0,10000)for all of the parameters α, β, γ and δ. For the standard deviations of the parameters we already used very vague priors, but those can be made even more vague. Let's change the prior for the common variance of the years also to Half-Cauchy(0,10000). Now the priors are very weakly informative, so practically all of the information is derived from the data.

Let's run the our hierarchical model again using these new priors, but using all the same specifications otherwise.

```
fit_alt_prior <- stan(file="hierarchical_TPH_AltPrior.stan", data=stan_data, iter = 2000,
            warmup = (2000/2), refresh=0)
draws_alt_prior <- data.frame(extract(fit_alt_prior))
```

Let's then compare the obtained parameters:

```
summary_alt <- summary(fit_alt_prior, probs=c(),
        pars=c("alpha","beta","gamma", "delta"))
summary_3$summary
```

```
##                mean    se_mean         sd     n_eff      Rhat
## alpha[1]   543.65386  0.9941166   53.54340 2900.9300 1.0009270
## alpha[2]   693.93896  0.5555167   34.31588 3815.8906 0.9997428
## alpha[3]   844.55349  0.6276946   36.30192 3344.7409 1.0002666
## alpha[4]   792.68911  0.5880960   36.12249 3772.7588 0.9991762
## beta[1]   -555.45991  2.7127721  113.10361 1738.3062 1.0013083
## beta[2]   -577.04026  2.1931157  107.66347 2409.9814 0.9998723
## beta[3]   -656.06281  3.4948421  116.88297 1118.5294 1.0023883
## beta[4]   -557.40091  2.0611110  102.44070 2470.2578 0.9999293
## gamma[1]   -66.31538  0.8122625   23.98563  871.9858 1.0028407
## gamma[2]  -122.21826  0.3558682   15.08372 1796.5475 1.0006541
## gamma[3]  -106.99847  0.4394191   16.28630 1373.6846 1.0011591
```

```
## gamma[4]  -127.48891  0.3294793   15.41819 2189.8312 0.9995188
## delta[1]  7100.63282 73.3288666 2118.50562  834.6594 1.0035826
## delta[2] 13623.47506 31.6611738 1330.11863 1764.9269 1.0009096
## delta[3] 14166.04170 38.4745295 1374.90158 1277.0167 1.0013017
## delta[4] 12955.16652 25.7872007 1219.22138 2235.4077 0.9999412
```

```
summary_alt$summary
```

```
##                mean     se_mean         sd      n_eff     Rhat
## alpha[1]   551.20921   5.2169482   55.83320  114.53865 1.037892
## alpha[2]   697.47056   1.7091488   32.22341  355.45391 1.014796
## alpha[3]   843.23617   1.7931442   35.35080  388.65839 1.019681
## alpha[4]   792.85412   1.2592895   34.58138  754.10801 1.005216
## beta[1]   -525.37456  22.5340720  133.40326   35.04722 1.092115
## beta[2]   -578.50082  10.2550201  113.94252  123.45213 1.033170
## beta[3]   -644.49667  11.3097579  120.04561  112.66421 1.035385
## beta[4]   -550.27701   2.4781157   99.21102 1602.79020 1.008143
## gamma[1]   -72.38068   3.5169823   26.36109   56.18060 1.054090
## gamma[2]  -120.68004   0.4412067   14.21338 1037.79280 1.006860
## gamma[3]  -106.76339   0.5272453   15.77443  895.12164 1.009375
## gamma[4]  -126.02901   0.8372238   15.37435  337.21783 1.010567
## delta[1]  7493.20888 225.8346898 2174.45214   92.70825 1.035581
## delta[2] 13459.08604  60.3495593 1274.09546  445.71344 1.013528
## delta[3] 14162.04040  76.0000781 1371.38510  325.60478 1.018305
## delta[4] 12857.25898  67.7157545 1237.84476  334.15882 1.010977
```

From the parameters we can see that they are extremely close to each other. This suggests that the weakly informative priors we chose were mostly overpowered by the data and didn't affect the results much at all. Using nearly flat priors produces results that are very similar to the ones we obtained when using weakly-informative priors.

Let's compare the models some more:

```
log_lik <- extract_log_lik(fit_alt_prior, merge_chains = FALSE)
r_eff <- relative_eff(exp(log_lik), cores = 2)
loo_alt <- loo(log_lik, r_eff = r_eff, cores = 2)

loo_compare(list(loo_3,loo_alt))
```

```
##        elpd_diff se_diff
## model1  0.0       0.0
## model2 -0.8       0.3
```

From the comparison, we can see that our estimated priors only slightly overperform the very weak priors. This suggests that the model is not very sensitive to the chosen priors as long as they are not made to be too limiting.

Discussion

Like all models, our model has some issues. One of the first issue was discovered during the data collection phase. While collecting the data for the daily precipitation numbers, we noticed that FMI only reports the

precipitation up to a single decimal. This inaccuracy in the data caused some binning especially at the lower ends of the precipitation amounts, which made fitting a linear model more difficult. We ended up applying a logarithmic scale to the precipitation data, which improved the accuracy of the models, but in the process we inevitably had to slightly alter the data to avoid $log(0)$ expressions.

Another problem with the weather data was that we have to assume that the weather for all trips matches the weather observations collected at the FMI weather station in Kaisaniemi. The Helsinki region is a large area, where there are bound to be some differences in weather at any given moment. The decision to only use data from the Kaisaniemi weather station was made purely for practical reasons. Trying to find and match more local weather data for the trips would have been very time consuming compared to the gain it would have provided for the model.

Another difficulty we had was with choosing the priors for the different models. There have been some previous studies conducted in other cities about the effect weather has on city biking, but none of their methods matched ours, which made it difficult to use their results for the priors. Instead we ended up looking to previous research for the expected directions of the effects as those were easily transferable to our study as well. With more time we could possibly try to use some more advanced methods to get our priors to reflect the prior research more accurately.

One issue also arose when we were choosing the models and especially the variables. During modeling we came up with many possible variables that could improve the predictive power of our model even further (f.e. workday vs holiday or what weekday it is). We also considered modeling different parameters of the biking dataset, such as duration and distance. However, to limit the scope of this project, we decided to stick to just weather variables and out of those to the three that we think have the largest effect, and the total trip number per day. If we were to continue this project further we would definitely look into including more variables.

We noticed that even though the hierarchical model performed best, sometimes the Markov-chains behaved slightly strangely, where they sometimes seemingly got "trapped" in a smaller range for several iterations. However, the hierarchical model is still clearly better than the pooled models.

Another observation came to our mind at the later stages of the project when we were thinking about the practical applications of the project. In the project we are using weather observations instead of weather forecasts. Since the traffic control decisions need to be made well in advance need to be based on forecasts. This means that the type of data that was used during the construction of the model (weather observations) does not exactly match the type of data that the model would be used with (weather forecast). It would be an interesting avenue of further research to see whether the results change significantly if we were to use data based on weather forecasts instead of observations.

Conclusion of the results

Our results confirm that weather has a significant effect on how likely people are to use city bikes in the Helsinki region. The results suggest that most trips are made on hot days, with no rain and low relative humidity. As weather can be forecast, HSL could use this model we created to allocate resources more effectively by adjusting the their workforce to match the expected demand on city bikes.

Self reflection on learning

During the project we learned especially about applying the topics of the course in practice. Even though many of the assignments have had real world examples, they have also had clear and guided answers for the questions. In the case of this project, many of the answers were not clear cut. Questions like which variables to include, what kind of a model should we build or what kind of a prior to choose are all questions that can have many justifiable answers depending on the circumstances. This project helped us think about the different choices we are making during the modeling and how do they actually affect the result.

This project also taught us about conducting data analysis in a team. At the early stages the distribution of the tasks was not as efficient as it could have been and we ended up waiting for each other or working on the same issues simultaneously without collaboration. However, as the project progressed, we started to find working methods that suited us and the project started to progress at a much greater speed.

References

Flynn, B. S., Dana, G. S., Sears, J., & Aultman-Hall, L. (2012). Weather factor impacts on commuting to work by bicycle. Preventive medicine, 54(2), 122-124.