Joel Schaniel
Badstrasse 1
5408 Ennetbaden
j.schaniel@yahoo.com

Data Science Project

Predicting Annual Bicycle Traffic from Short-Term Measurements

Conceptual Design Report

05. October 2025

Abstract

This project develops a predictive model to estimate the average daily number of cyclists at any location in the canton of Zurich based on short-term measurement periods of up to seven consecutive days. The goal is to significantly increase the value of such short surveys by incorporating weather data as well as seasonal and weekly patterns. Bicycle count data is sourced from the Swiss open data portal, while weather data is obtained from MeteoSwiss. Supervised learning methods such as Random Forests and Boosted Decision Trees are used for model training. Data processing is performed using Python. The model aims to minimize the influence of weather-related fluctuations on the counts, enabling more reliable extrapolation to annual values. The big data basis set a good foundation for reliable results. However, the impact of other parameters and disturbances like construction works are unclear. The entire code will be made available in a GitHub repository to ensure transparency and reproducibility. In the long term, the model has the potential to optimize traffic planning and management while reducing data collection costs.

Table of Contents

1 Project Objectives

1.1 Background

One of the most valuable characteristics in most traffic studies are traffic counts. Counts are measured in expensive and time-consuming traffic surveys. Due to budget and time constraints, often short sample surveys are used instead. This raises the problem that the periods being evaluated should be as representative as possible. In motorized private transport, such sample surveys can be interpreted fairly well based on seasonality and days of the week. However, bicycle traffic is much more variable and particularly dependent on the weather.

A reliable model that estimates yearly cycling volumes from a short sampling period, including weather parameters could increase the value of such short measurements and enhance decision-making.

1.2 Objective

This project aims to develop a predictive model that estimates the average annual number of cyclists at any location in Switzerland, based on short-term measurement periods of just up to 7 consecutive days. The model incorporates counting data of every permanent bicycle counting station in the canton of Zürich and combines it with weather data and seasonal patterns to generalize annual trends from minimal data.

The model can be used to adjust data from short measurement periods under consideration of the aforementioned inputs to the annual average.

2 Methods

2.1 Infrastructure

The project is implemented using Python 3.12 on a Google Colab infrastructure. A combination with a Google Drive repository allows for an uncomplicated integration of the data in the code.

2.2 Software libraries

The following software libraries are intended to be used for the project:

1)	Requests [12]	Web scrapping - data retrieval
2)	BeautifulSoup [13]	Web scrapping - parsing
3)	Pandas [15], numpy [8]	Data manipulation, cleaning, and numerical operations.
4)	GeoPandas [10]	Data wrangling for spatial data
5)	Statsmodels [14], SciPy [16]	Data exploration and statistical tests
6)	Scikit-learn [11]	tools for supervised learning
7)	Matplotlib [9], seaborn [17]	Data visualisation

2.3 Supervised Learning

The model is to be trained using supervised learning. Various algorithms will be tested for their suitability. In particular, random forests and boosted decision trees are to be tested.

3 Data

The project contains data from two different sources:

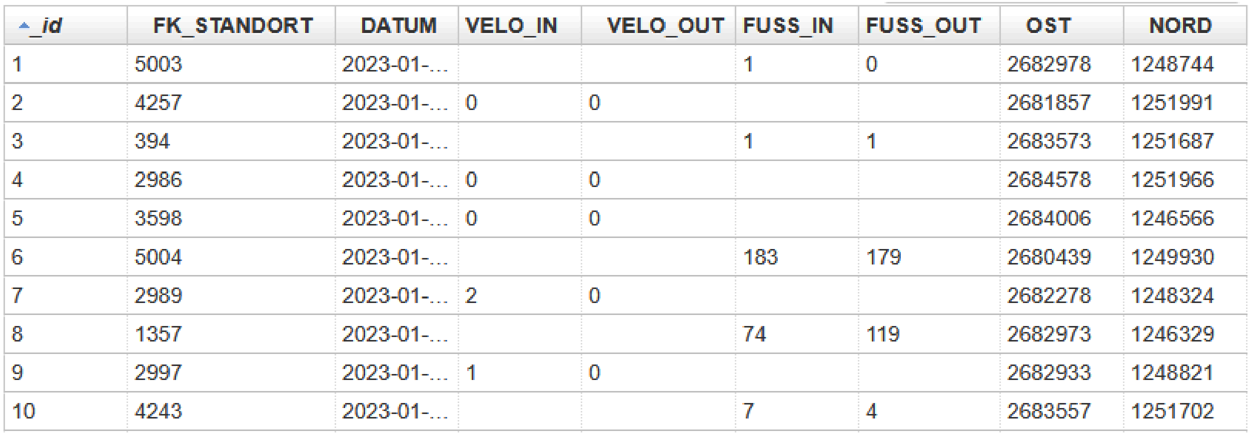
- Counting data from the Swiss open data portal [4] [5]
- Weather data from Meteo Schweiz [6]

3.1 Counting data

The counting data is not exactly identical in every canton, therefore this project restics to data from the canton of Zürich. The sample size is still large, to ensure a high model quality.

The counting data is publicly available and can be downloaded as CSV-files, aggregated to 15 minutes intervals. Besides a ID for each measurement, it offers a location ID, a time stap, counts for bicycles and pedestrians per direction and coordinates of the location.

Table 1: First 10 rows of the data set.



_id	FK_STANDORT	DATUM	VELO_IN	VELO_OUT	FUSS_IN	FUSS_OUT	OST	NORD
1	5003	2023-01-…			1	0	2682978	1248744
2	4257	2023-01-…	0	0			2681857	1251991
3	394	2023-01-…			1	1	2683573	1251687
4	2986	2023-01-…	0	0			2684578	1251966
5	3598	2023-01-…	0	0			2684006	1246566
6	5004	2023-01-…			183	179	2680439	1249930
7	2989	2023-01-…	2	0			2682278	1248324
8	1357	2023-01-…			74	119	2682973	1246329
9	2997	2023-01-…	1	0			2682933	1248821
10	4243	2023-01-…			7	4	2683557	1251702

3.2 Weather data

Weather data for Switzerland is also openly available. It is provided by Federal Office of Meteorology and Climatology MeteoSwiss (https://www.meteoswiss.admin.ch/services-and-publications/service/open-data.html). Temperature and precipitation data can be aggregated to daily values, which might be a reasonable aggregation for our purpose. For increased accuracy, only weather stations in the canton of Zürich are selected.

4 Metadata

Since all data sources are openly available to the public, every step can be reproduced without further metadata necessary. However, the actual python script, containing all necessary information like hyperparameters can be shared in a GitHub repository.

5 Data Quality

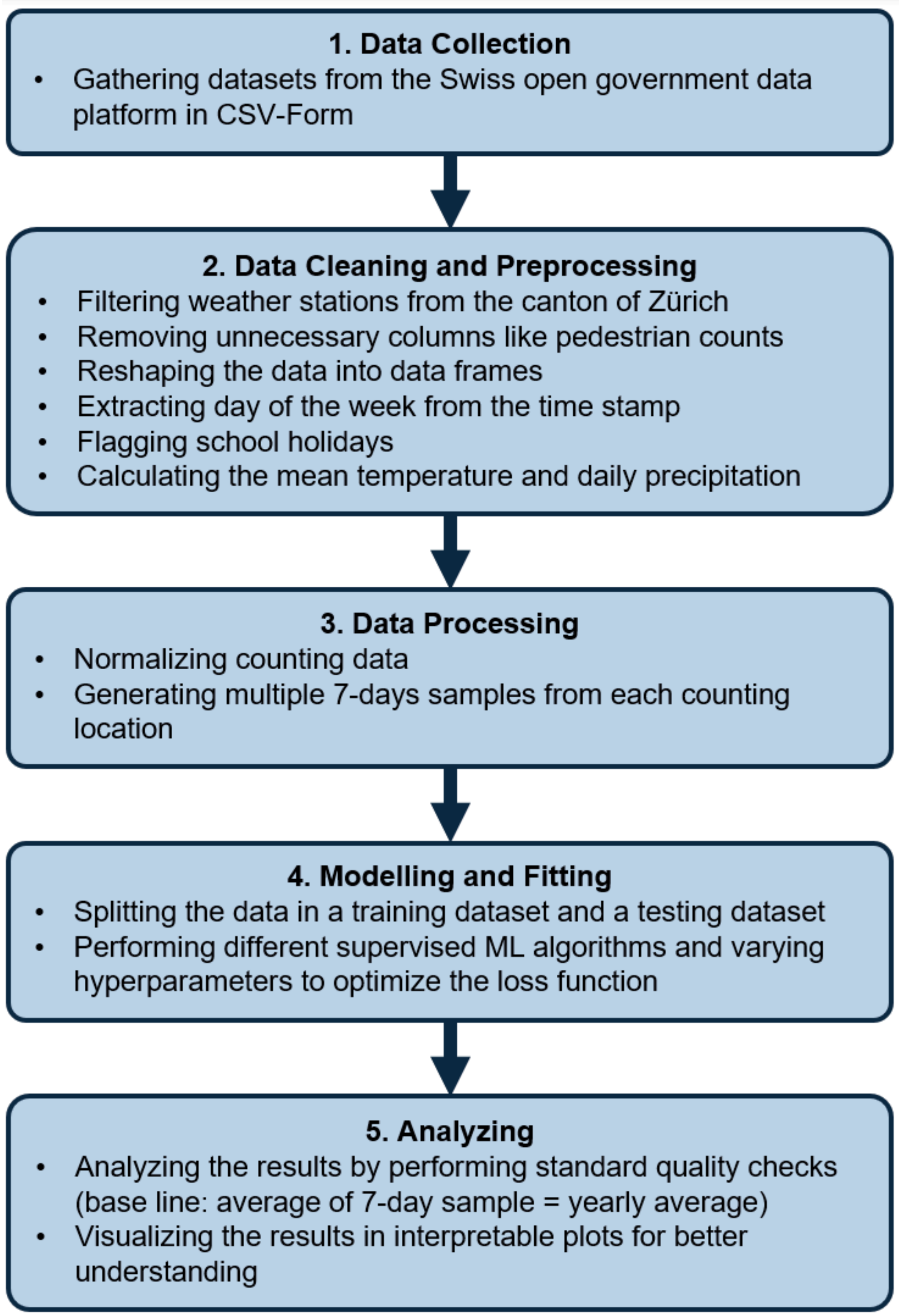
Our traffic count dataset is very large, which ensures a solid foundation. Missing values can simply be excluded with no impact on the model. However, traffic counts can be influenced by many factors that are not apparent in a cursory data analysis. The most common influences are accidents, construction work, detours and damaged sensors. Short-term disturbances, such as accidents, lead to outliers which only have a limited impact due to the large dataset. Long-term obstructions, such as large-scale construction work, might also have a limited impact, as we expect the same weather dependencies for these routes as for others. However, construction work or detours lasting only a few weeks or months (for example, during the summer) could negatively impact our model. As the impact of such disturbances can be either positive or negative for our counts (e.g. construction on the road with the measurement point versus construction on the neighbouring street with a detour over the measurement point), we would expect higher variance in our model.

Improving the quality of our traffic count data would require manual inspection of all time series, which is not feasible. As only a small proportion of our data will be affected by disturbances, the quality of the model is still expected to be reasonable.

The data coverage and accuracy of the weather dataset are very good. Due to the small spatial extent of Zurich, local differences between weather stations should be negligible since they correlate heavily. For temperature data, the mean temperature across all weather stations can be used. Therefore, missing values from individual weather stations can be disregarded. The canton of Zurich has no significant elevation differences that would result in large temperature variations between weather stations.

Precipitation data is gathered at 15-minute intervals. The mean daily precipitation sum can be used for modelling. If values from one or more 15-minute intervals are missing, the daily sum from these weather stations can be excluded from the calculation of the mean.

6 Data Flow



1. Data Collection

- Gathering datasets from the Swiss open government data platform in CSV-Form

2. Data Cleaning and Preprocessing

- Filtering weather stations from the canton of Zürich
- Removing unnecessary columns like pedestrian counts
- Reshaping the data into data frames
- Extracting day of the week from the time stamp
- Flagging school holidays
- Calculating the mean temperature and daily precipitation

3. Data Processing

- Normalizing counting data
- Generating multiple 7-days samples from each counting location

4. Modelling and Fitting

- Splitting the data in a training dataset and a testing dataset
- Performing different supervised ML algorithms and varying hyperparameters to optimize the loss function

5. Analyzing

- Analyzing the results by performing standard quality checks (base line: average of 7-day sample = yearly average)
- Visualizing the results in interpretable plots for better understanding

Figure 1: Process diagram of the project. [1], [2], [3]

7 Data Model

7.1 Conceptual level

The data model minimizes the influence of weather parameters on traffic counts. Using long time series for each counting location, the model identifies the impact of weekdays, holiday periods, precipitation, and temperature on bicycle traffic. The trained model enables the use of counting data regardless of weather conditions during the measurement period, resulting in more affordable and reliable traffic surveys.

7.2 Logical level

On a logical level, the following data will be used to train a model with multiple supervised learning algorithm, including random forest and boosted trees [3] .

Counting data:

- Date (and derived from the date: day of the week, season, holidays)
- Normalized counts, derived into multiple 7-days samples

Weather data:

- Date
- Average temperature
- Daily sum of precipitation

If the model's results are unsatisfactory, a clustering algorithm can be used to include additional features. For example, it would be sensible to identify different behaviors of commuting and leisure routes.

7.3 Physical level

The model runs on conventional computers. No special infrastructure is necessary.

8 Documentation

The code can be shared and documented in a GitHub repository. Well-commented code ensures the readability.

9 Risks

The following risks were considered:

- Data is no longer publicly available: This is only a long-term risk since updating the model with new data is only necessary if major mobility trends shift cyclists' behavior. Furthermore, it is highly unlikely that the authorities will reverse their open-access philosophy.
- High variance: It is possible that other unidentified parameters significantly impact the traffic counts. This would lead to high variance in the model and worse results. One possible solution is to use unsupervised learning and clustering to identify different clusters and reverse-engineer the missing parameters with a significant impact.
- Inexperience: Since this is my first real machine learning project, implementing and optimizing the algorithms may take a lot of time. An additional risk is that misbehavior of the model might not be detected immediately.

10 Conclusions

Open data enables the development of powerful applications for traffic management. The described model could improve the quality of traffic surveys while reducing costs. However, it is unclear whether temperature and precipitation can account for most of the variation in daily counts. Many details can only be determined within the project; therefore, it is too early to draw a conclusion.

One additional approach to the project could be examining time series models.

Statement

„Ich erkläre hiermit, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls die Arbeit als nicht erfüllt bewertet wird und dass die Universitätsleitung bzw. der Senat zum Entzug des aufgrund dieser Arbeit verliehenen Abschlusses bzw. Titels berechtigt ist. Für die Zwecke der Begutachtung und der Überprüfung der Einhaltung der Selbstständigkeitserklärung bzw. der Reglemente betreffend Plagiate erteile ich der Universität Bern das Recht, die dazu erforderlichen Personendaten zu bearbeiten und Nutzungshandlungen vorzunehmen, insbesondere die schriftliche Arbeit zu vervielfältigen und dauerhaft in einer Datenbank zu speichern sowie diese zur Überprüfung von Arbeiten Dritter zu verwenden oder hierzu zur Verfügung zu stellen.“

Date: 05.10.2025

Signature(s):

References and Bibliography

[1] A. Mühlemann et al. (2025). *CAS ADS Lecture Slides Module 1.*

[2] A. Mühlemann et al. (2025). *CAS ADS Lecture Slides Module 2.*

[3] A. Marcolongo et al. (2025). *CAS ADS Lecture Slides Module 3.*

[4] EcoCounter (2025). *Velozähldatenzentrale Schweiz.* https://velo-ch.eco-counter.com/

[5] Schweizerische Eidgenossenschaft et al. (2025). *Swiss Open Government Data platform.* https://opendata.swiss/de

[6] Federal Office of Meteorology and Climatology (2025). *Ground-based measurements and homogeneous data series from individual stations.* https://www.meteoswiss.admin.ch

[7] OpenAI. (2025). *ChatGPT* (Oct 05 version) [Large language model]. https://chat.openai.com/chat

[8] Harris, C. R., et al. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

[9] Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

[10] Jordahl, K., et al. (2023). *GeoPandas: Python tools for geographic data* (Version 0.14.0) [Computer software]. https://geopandas.org/

[11] Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://www.jmlr.org/papers/v12/pedregosa11a.html

[12] Reitz, K., & contributors. (2023). *Requests: HTTP for humans* (Version 2.31.0) [Computer software]. Python Software Foundation. https://docs.python-requests.org/

[13] Richardson, L. (2023). *Beautiful Soup: Screen-scraping library* (Version 4.12.2) [Computer software]. https://www.crummy.com/software/BeautifulSoup/

[14] Seabold, S., & Perktold, J. (2010). Statsmodels: Econometric and statistical modeling with Python. In *Proceedings of the 9th Python in Science Conference* (Vol. 57, pp. 92–96). https://www.statsmodels.org/

[15] The pandas development team. (2023). *pandas* (Version 2.1.0) [Computer software]. https://pandas.pydata.org/

[16] Virtanen, P., et al. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, *17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2

[17] Waskom, M. L. (2021). Seaborn: Statistical data visualization. *Journal of Open Source Software*, *6*(60), 3021. https://doi.org/10.21105/joss.03021