Wifi-fingerprint positioning

Evaluating technics for indoor positioning based on Wifi-fingerprinting

Edison Guevara Bastidas

Outline

Wifi-fingerprint positioning

1. Motivation
2. Objectives and goal
3. Dataset
4. Pre-processing of the data
5. Model selection
6. Final error metrics
7. Next-steps

Motivation

Wifi-fingerprint positioning

Many real world applications need to know the localization of a user in the world to provide their services.

Indoor localization is still an open problem mainly due to the loss of GPS signal in indoor environments.

WiFi based positioning doesn't require additional hardware infrastructure

Some other technics are:

- Bluetooth
- IR
- Magnetic fields
- Ultra sonic waves

Objectives and goal

Wifi-fingerprint positioning

Objectives

- Investigate the feasibility of using “wifi fingerprinting” to determine a person's location in indoor spaces.
- Evaluate multiple machine learning models to see which produces the best result

Goal

- Make a recommendation as whether WiFi-fingerprinting should be used for indoor positioning

Dataset

Wifi-fingerprint positioning

Training set:

19937 datapoints

933 unique locations

Validation set:

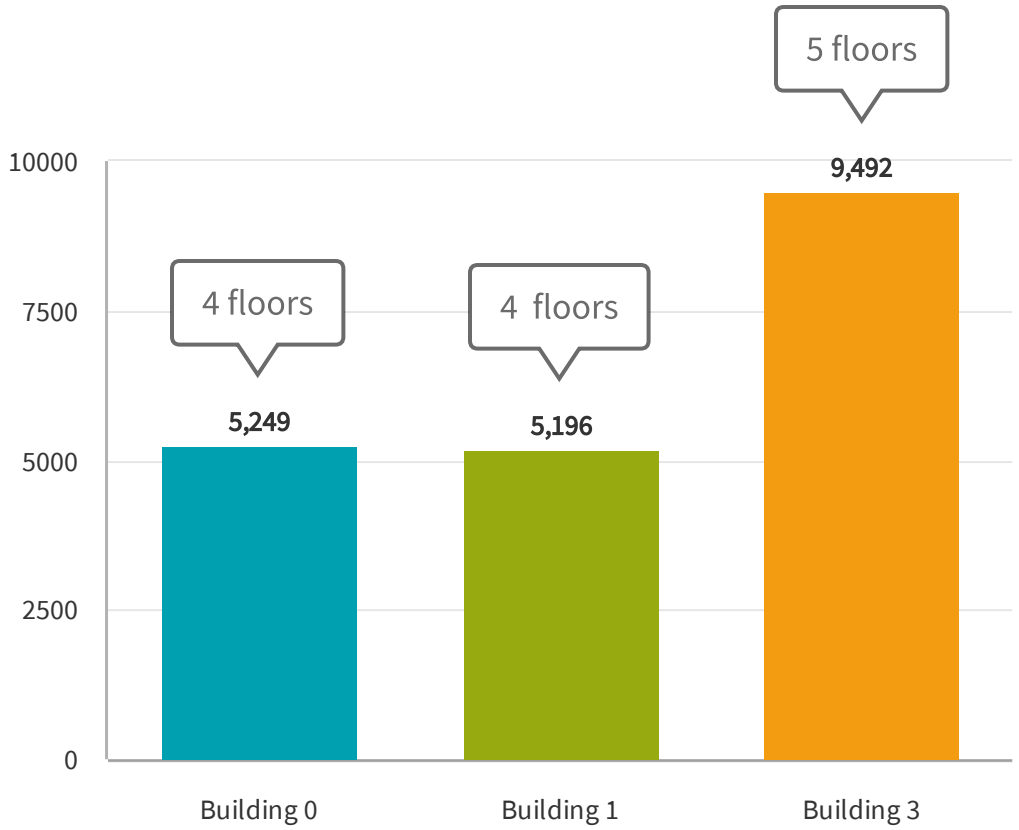
1111 datapoints

1111 unique locations

Datapoints:

RSSI range: [-104...0]

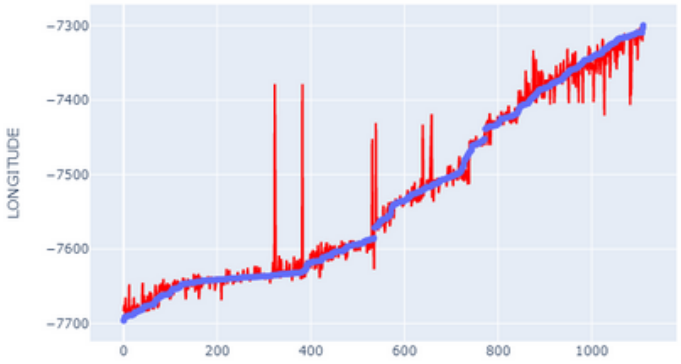
No signal = 100



Although the training set has 20 times more data than validation, it has similar number of unique locations -> Challenge!

Pre-processing

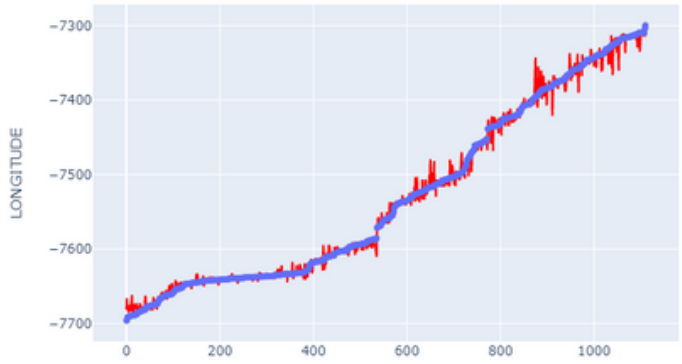
Wifi-fingerprint positioning



100 replaced as -200

linear transformation: offset +105

unit norm normalization



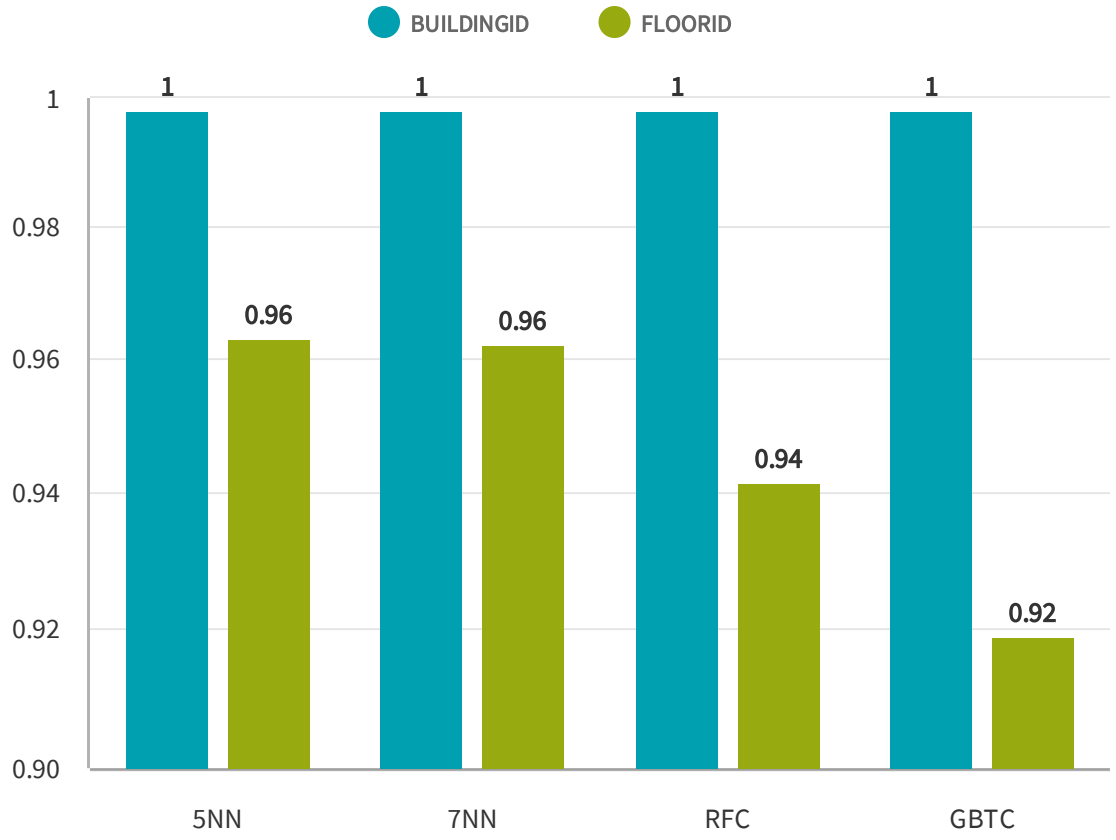
transformation: 10^(P(dBm)/ (n·10))

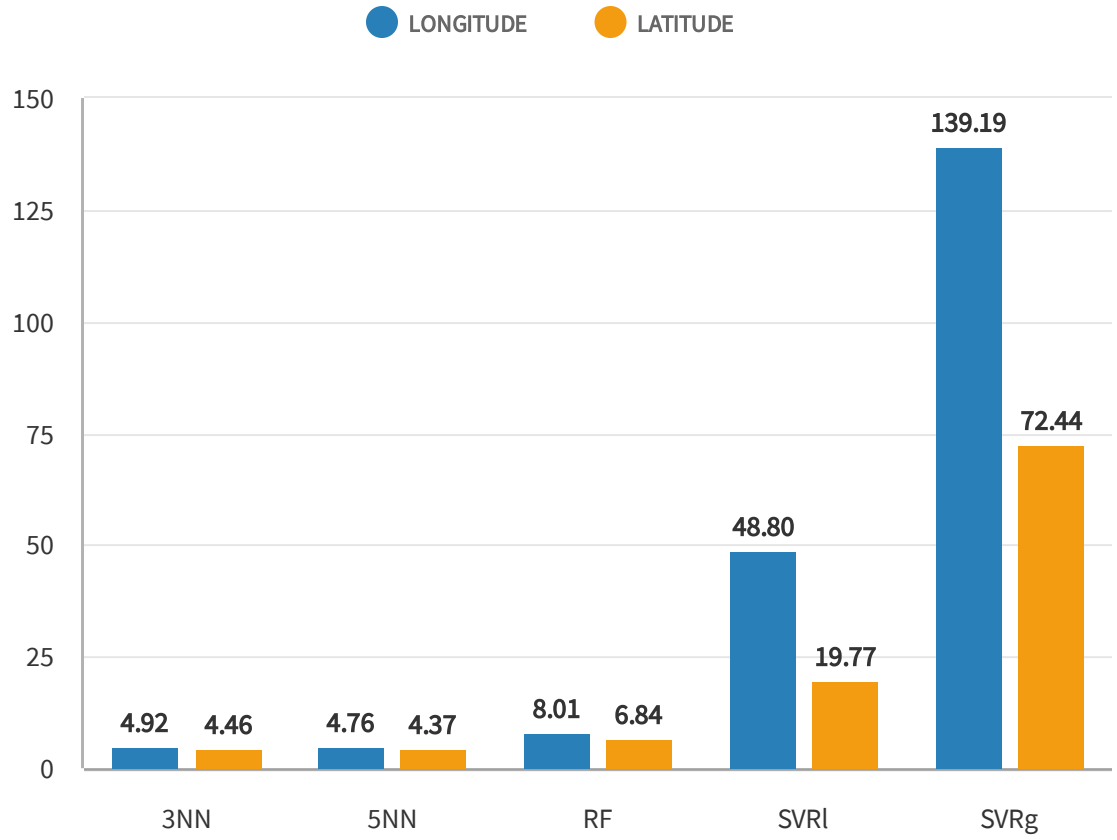
unit norm normalization

The power transformation can cope better with max error. Values of n = [2, 3, 4, 5, 6] have been explored -> 4 and 5 best values

Model selection

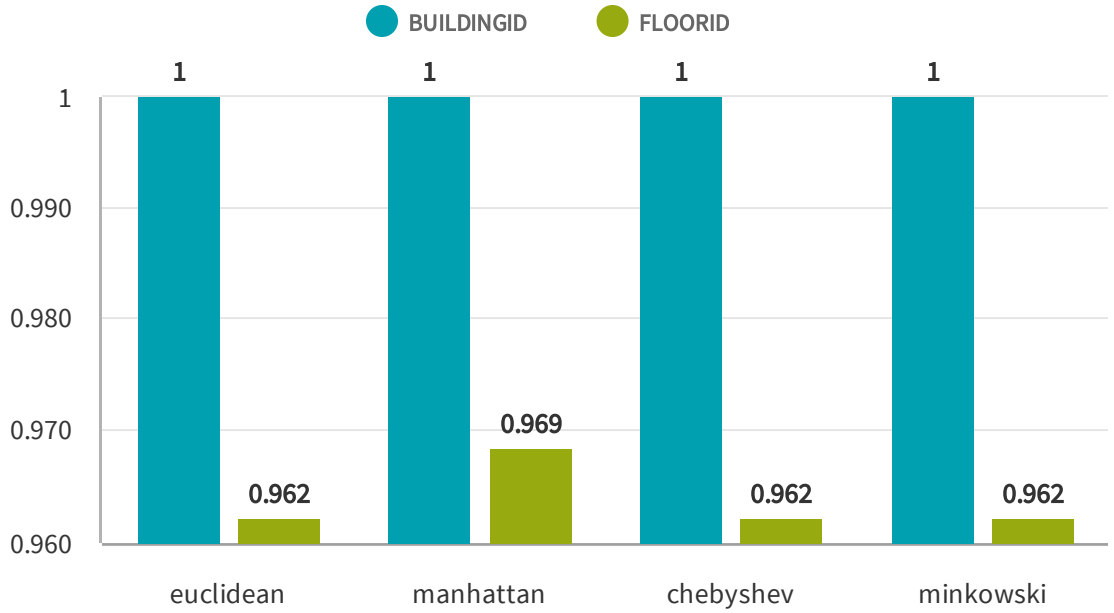
Wifi-fingerprint positioning

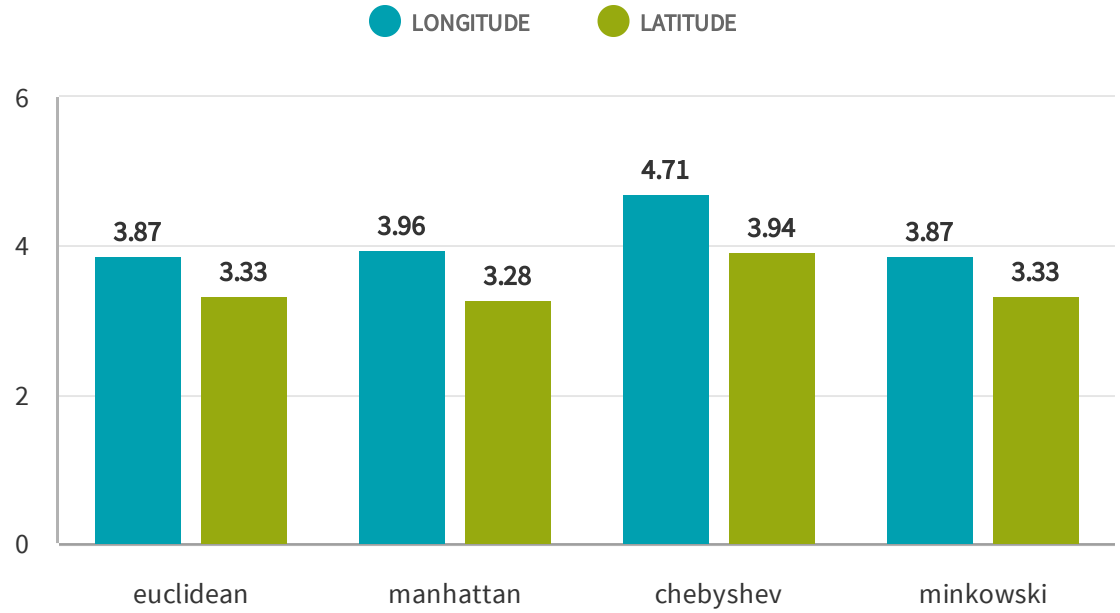




Model selection - 5NN hyper-parameters - distance metric

Wifi-fingerprint positioning

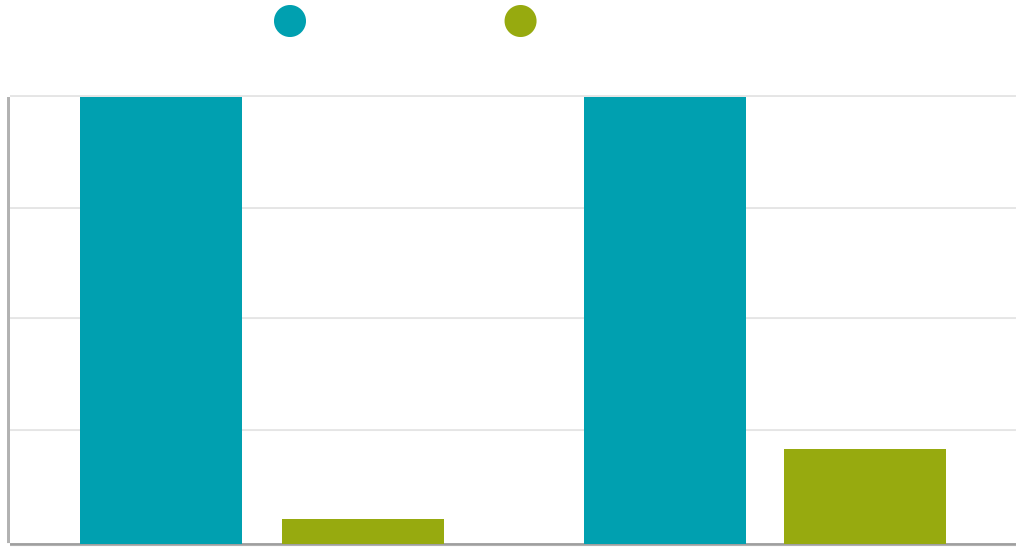


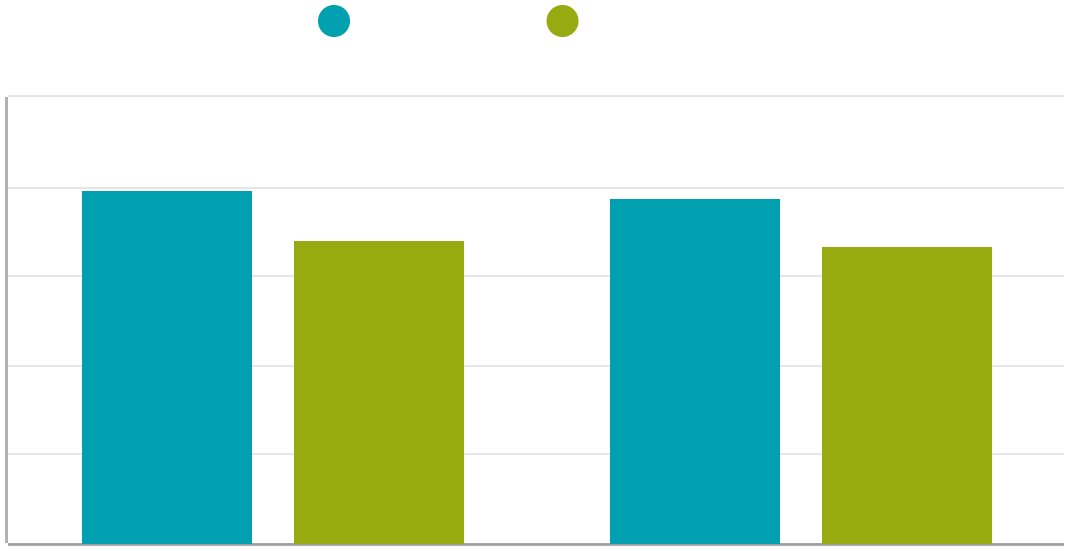


MAE for building 0 for different distance metrics

Model selection - 5NN hyper-parameters - weighted distance

Wifi-fingerprint positioning

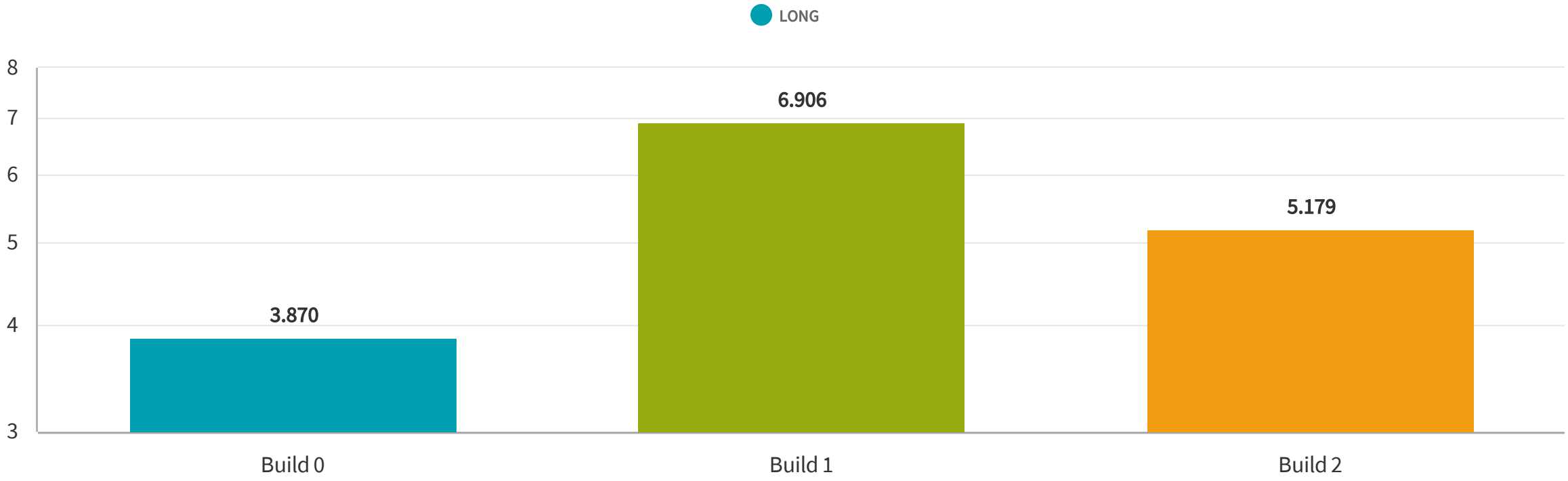




MAE for building 0 for different weight

Model selection - 5NN - split by building

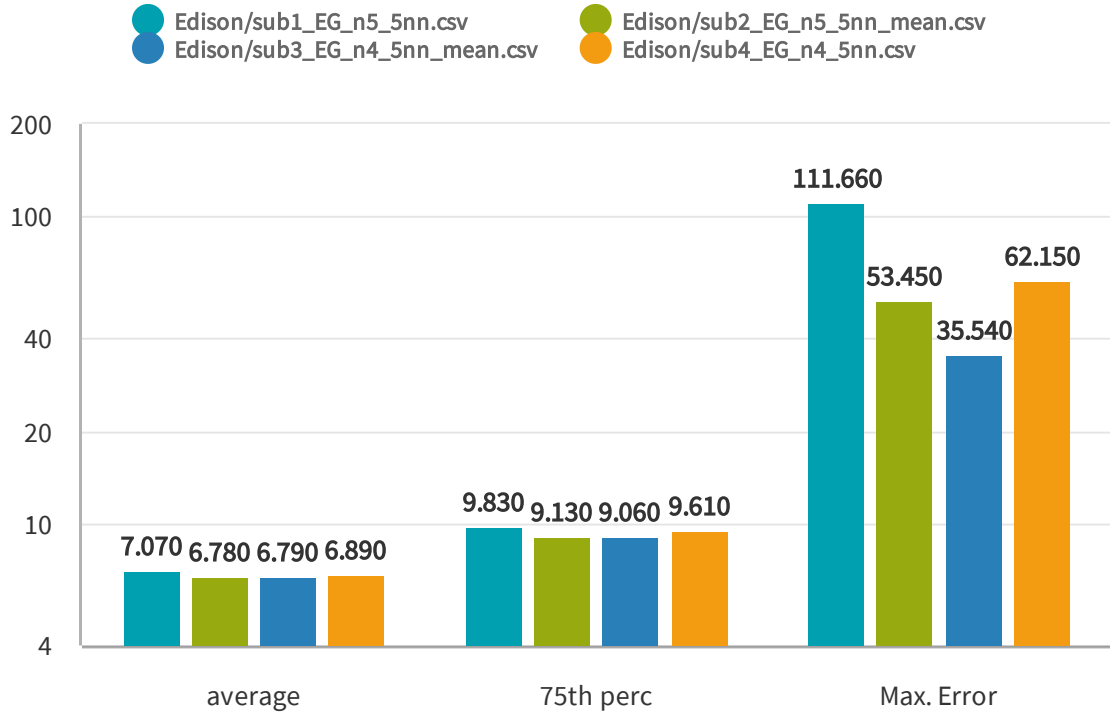
Wifi-fingerprint positioning

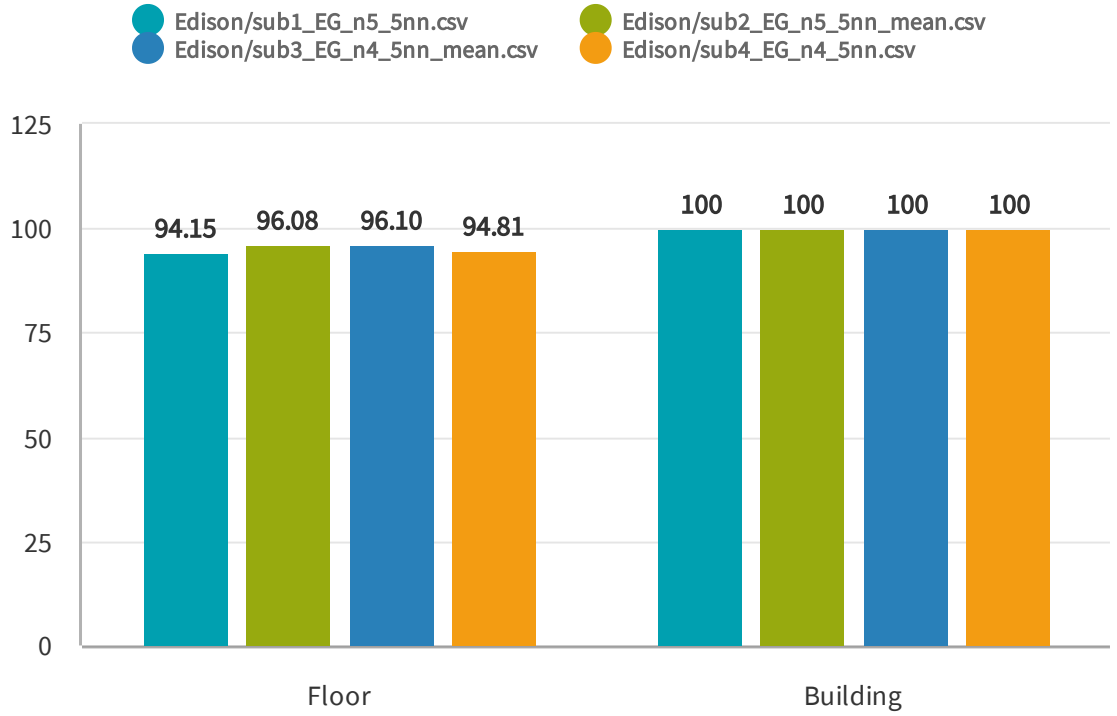


Error in building 0 is lower than the other two

Final error metrics

Wifi-fingerprint positioning





MAE for building 0 for different weight

Next-steps

Wifi-fingerprint positioning

1 Lessons learnt

- Cross validation is useful to have relevant error metrics and avoid over fitting and generalize the prediction

- Buildings 1 and 2 are more difficult to predict. New strategies are needed here.

2 Next-steps

- Filter out outliers could improve metrics in general and specific in buildings 1 and 2

- bootstrap in order to get confidence intervals on the error metrics

- Include additional features

Questions?