Experiments

* **Hypothesis:** different frequency data has different seasonality and trend structures. I expect that models’ performance will be different on data of different frequencies.
  1. How different frequencies affect results?
     + **Why?** Time-series data can have different frequency and it makes sense that the same data of different frequency behaves in a different way – hence it is expected that TSFMs will perform differently on the data of different frequency.
     + **Solution:** Run experiment on data from the same “source” but of a different frequency
* **Hypothesis:** different frequency data has different seasonality and trend structures. I expect that models’ performance will change if we use different context lengths and how the performance changes depends on the frequency of data.
  1. How different context length affects results?
     + **Why?** Because depending on what kind of data we want to use the model for, we want to know which context length to use.
     + **Solution:** Experimenting with different fine-tuning lengths and different frequencies
* **Hypothesis:** different frequency data has different seasonality and trend structures. I expect that models’ performance will change if we use different context lengths and how the performance changes depends on the frequency of data.
  1. How the combination of context length and frequency affects results?
     + **Why?** Because depending on what kind of data we want to use the model for, we want to know which context length to use.
     + **Solution:** Experimenting with different fine-tuning lengths and different frequencies
* **Hypothesis:** It is important that the TSFM is fine-tuned with the most recent data because trends/patterns in data constantly change, so if we fine-tune on older data, TSFM will not perform as well.
  1. How does having “train” set inside the fine-tuning set affect TSFM performance?
  2. How does the “lag” of the fine-tune data impact TSFM performance?
     + (having “train” set inside the fine-tune set == finetuning on most recent data)
     + (having “train” set separated from the fine-tune set == finetuning on “further” data: the “lag” of the fine-tune data is equal to the context length)
     + (train-in-finetune vs train-out-finetune versions should be compared with the came context length)
     + **Why?** Becaue having fine-tune data being close to the time of prediction means that the TSFM can learn the most recent pattern. With a gap between fine-tune data and prediction time, TSFM could potentially learn a patternt which is no longer relevant and would hence be “confused”.
     + **Solution:** Experiment with having the “train” set inside the fine-tune data and having them separated. In cases where they are separated, experiment with different context lengths.
* **Hypothesis:** TSFMs perform differently on different types of data (relative to benchmark)
  1. Which type of financial data do TSFMs perform best?
  2. Why do TSFMs perform better on some types of financial data?
     + **Why?** Different types of data have different characteristics: seasonality, trend, order of stationarity, etc… Therefore it is logical to assume that TSFM will have different performance on different types of data.
     + **How?** Run experiments across different types of data. Classify types of data according to their characteristics – seasonality, trend, etc. and analyse TSFMs performance according to these data characteristics.
       - **Stock price (index)**
       - **Stock return**
       - **Credit card**
       - Credit card volumes
       - Commodity prices

(potential hypothesis: play around with prediction horizon – only for cc volumes and commodity prices)

* **Hypothesis:** TSFMs will perform better on the “prominent” financial data from before February 2024 than on the same data after February 2024.
  1. Do TSFMs perform better of Stock index prices from before Feb 2024 than they do on Stock indexes from after Feb 2024?
  2. To what degree do TSFMs “remember” the data they were trained on?
     + **Why?** TSFMs were probably trained on Stock index prices from before Feb 2024 therefore they probably could remember the actual values from their pre-training
     + **How?** Run experiments on data before and after Feb 2024 and calculate the difference in TSFM performance
       - With and without finetuning
* Repeat the TSCV across different periods of the time-series
  1. No need to try out for different number of folds and different number of times TSCV is repeated across a single time-series. I should pick a number for both and stick with that
  2. Make sure that the TSCV runs are equally spaced out throughout the time-series
* Stock data vs return data
  1. stock data results will probably be bad so put them in appendix along with other results that turn out bad
  2. Inquire about and use different time-series preprocessing methods that Ramin mentioned
     + Differencing, etc…
* When is the data from?
  1. Use data after February 2024 to avoid data leakage.
  2. Including data before February 2024 could cause data leakage but it would be interesting to compare TSFM’s performance with and without leakage
     + In order to maximize the chances of leakage, we should use index prices because they are most likely to have been included in training
     + Tie this to the concept of “machine unlearning”
       - Talk about “machine unlearning” in the Future Work section
* Consider other types of data
  1. Exchange rates
  2. Commodities
  3. Crypto
  4. Credit card data (from Lending Club)
* Repeat the whole experiment multiple times (seats)