30.04.25

OPENCAMPUS.sh

# Advanced Time Series Prediction

**WEEK 3**

- Organizational Matters
- Week 3:
  - Signals
  - Fourier Tansform
  - Kalman-Filtering and State-Spaces

# ORGANIZATIONAL MATTERS

- **Course Projects: How is it going?**
- **Please be prepared**
- **Anything else?**

# TASKS UNTIL NEXT WEEK

- Watch the videos for week 4
- Do the Exercises
- Projects:
  - PLOT YOUR DATA !!!
  - PLOT YOUR DATA !!!
  - PLOT YOUR DATA !!!

# EXERCISES

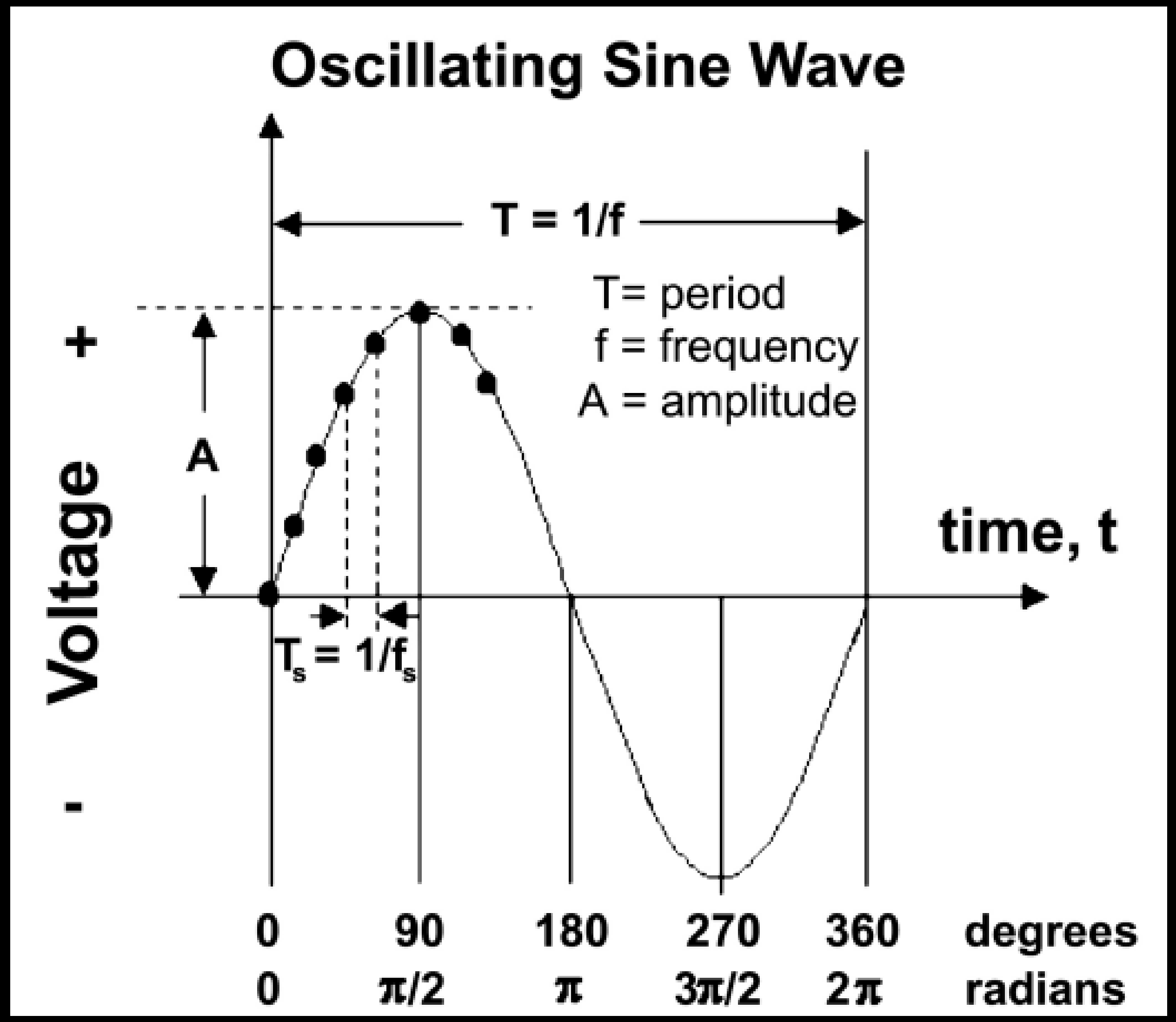
- Each of you presents at least once:
  - If you need to repeat some basics:
    - Trees1_Linear_Regression.ipynb
    - Trees2_The_Decision_Tree.ipynb
  - Every group prepares these notebooks:
    - Trees3_The_Random_Forest.ipynb
    - Trees4_Gradient_Boosting_with_XGBoost_and_LigthGBM.
    - Trees5_CatBoost.ipynb

# QUIZ:

- What is the log power spectrum?
- What is the phase of a signal?
- What is np.fft.fft?
- Who is/was Kalman?
- What is meant by filtering?
- For which purpose would you use it?

# SIGNALS:

- A time series signal can be broken down into a weighted sum of sine waves.
  - Characterized by amplitude, frequency and phase
- The power spectrum is a plot of the amplitude of each sine wave component against its frequency.
  - It shows the relative importance of each frequency component to the overall signal.
- The log power spectrum is created by taking the logarithm of the power spectrum.
  - It tells us the relative importance of each frequency component to the overall signal.
  - Very often you might see something like this:
    - 100 Pascal ≈ 134 dB ≈ 676 sone
    - Amplification or reduction of signal strength



# FOURIER TRANSFORM:

- The Fourier Transform is a mathematical tool that decomposes a signal into its constituent frequencies.
  - This is analogous to separating mixed paint colors back into their original components.
- It can be represented as a complex number, with the real and imaginary components reflecting the x and y coordinates of the center of mass respectively.
- It is widely used in various fields, including sound editing, where it can be used to identify and remove unwanted frequencies from a recording.
  - Or in our case "cleaning" of a time-series signal

$$X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt$$

# STATE-SPACE REPRESENTATION:

1) To apply the Kalman filter, we need a mathematical model of the system we're observing - State-space representation provides a convenient framework for this task
   - In this representation, we use state variables to fully characterize the system at any given time. For example, a robot's state could be defined by its position and velocity
   - State-space equations describe how the state variables evolve over time based on the system's dynamics and inputs. These equations can be expressed in both continuous and discrete forms, the latter being crucial for digital implementations
2) Quantifying Uncertainty with Gaussian Distributions
   - Real-world sensor measurements always contain noise, adding uncertainty to our understanding of the system's true state
   - The Kalman filter embraces this uncertainty by modeling the noise as Gaussian distributions
   - In the one-dimensional case, a Gaussian distribution is defined by its mean and variance. The mean represents the most likely value, while the variance captures the spread or uncertainty around the mean.
   - For multi-dimensional systems, we use a mean vector and a covariance matrix to describe the multivariate Gaussian distribution.
   - The covariance matrix captures not only the individual variances of each state variable but also the correlations between them

# KALMAN FILTERING:

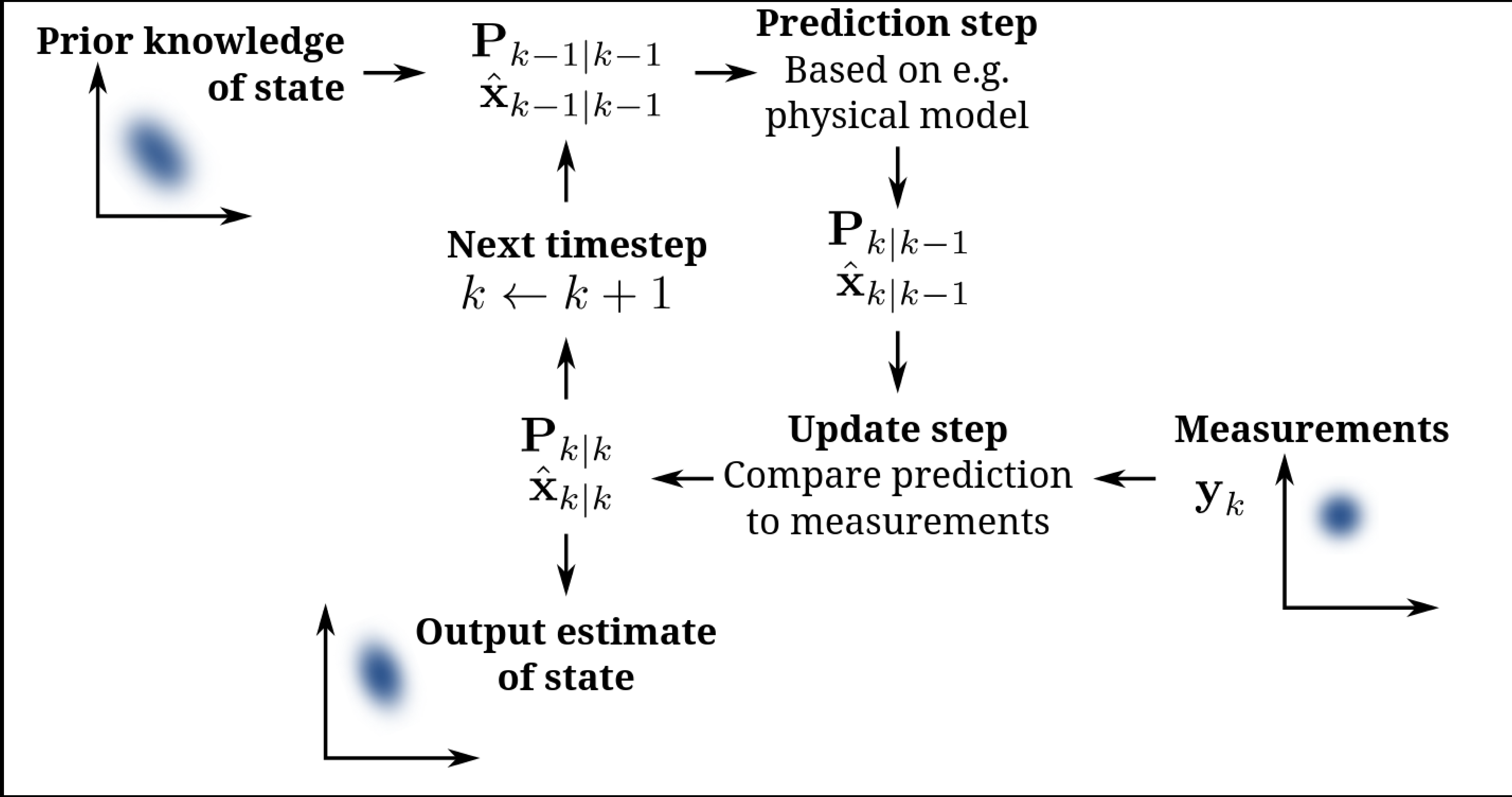
3. The Kalman filter operates in a two-step iterative process: prediction and correction

- Prediction Step:
  - Using the state-space model, we predict the system's future state based on our current best estimate. We also predict the corresponding covariance matrix, accounting for the inherent uncertainty in the model and the accumulation of noise over time
- Correction Step:
  - We obtain sensor measurements of the system's output. These measurements provide a new piece of information, independent of our model-based prediction. We then fuse the predicted state and the measured output, weighting them according to their respective uncertainties. This fusion process produces an updated, more optimal estimate of the system's state, along with an updated covariance matrix

4. The Kalman Gain: Balancing Prediction and Measurement

- It determines the relative weight given to the prediction and the measurement when fusing them to obtain the updated state estimate.
- The Kalman gain is calculated based on the covariance matrices of the prediction and the measurement. If the prediction is highly uncertain (high covariance), the Kalman gain will be larger, giving more weight to the measurement. Conversely, if the measurement is very noisy, the Kalman gain will be smaller, favoring the prediction

# KALMAN FILTERING:



# EXERCISES

- Each of you presents at least once:
  - If you need to repeat some basics:
    - Trees1_Linear_Regression.ipynb
    - Trees2_The_Decision_Tree.ipynb
  - Every group prepares these notebooks:
    - Trees3_The_Random_Forest.ipynb
    - Trees4_Gradient_Boosting_with_XGBoost_and_LigthGBM.
    - Trees5_CatBoost.ipynb

# TASKS UNTIL NEXT WEEK

- Watch the videos for week 4
- Do the Exercises
- Projects:
  - PLOT YOUR DATA !!!
  - PLOT YOUR DATA !!!
  - PLOT YOUR DATA !!!