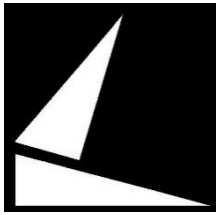
September 3rd-5th, 2018
INESC Technology and Science (INESC TEC)
Porto, PORTUGAL

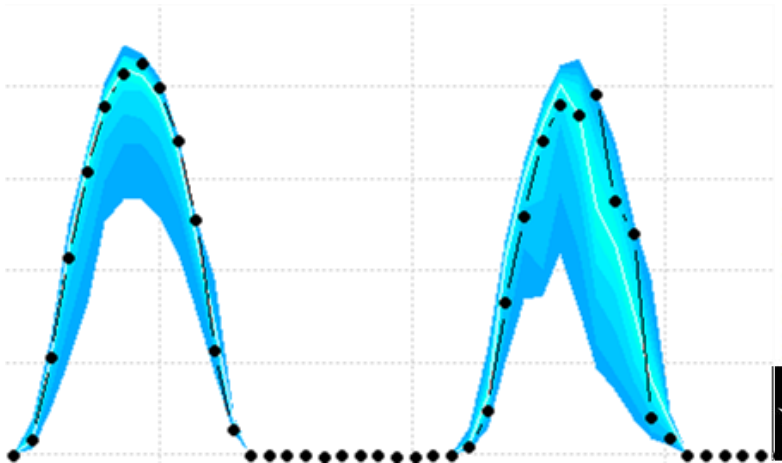


2018-04: Advanced Data Analytics for Energy Systems

Statistical learning for uncertainty forecasting

Jethro Browell
University of Strathclyde

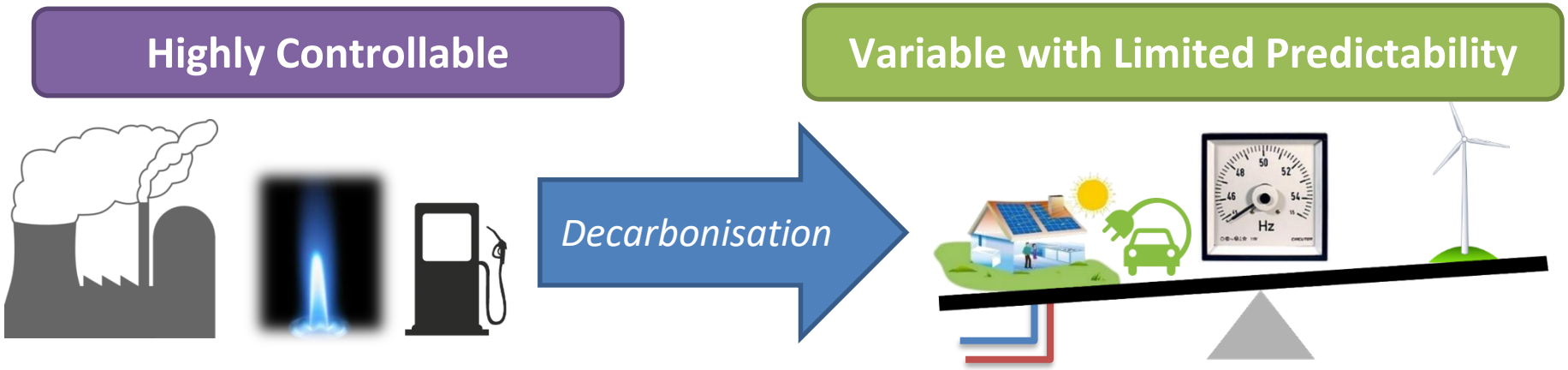
Contents



EES-UETP
Electric Energy Systems - University Enterprise Training Partnership

1. Uncertainty Forecasting

What is it and why should I care?

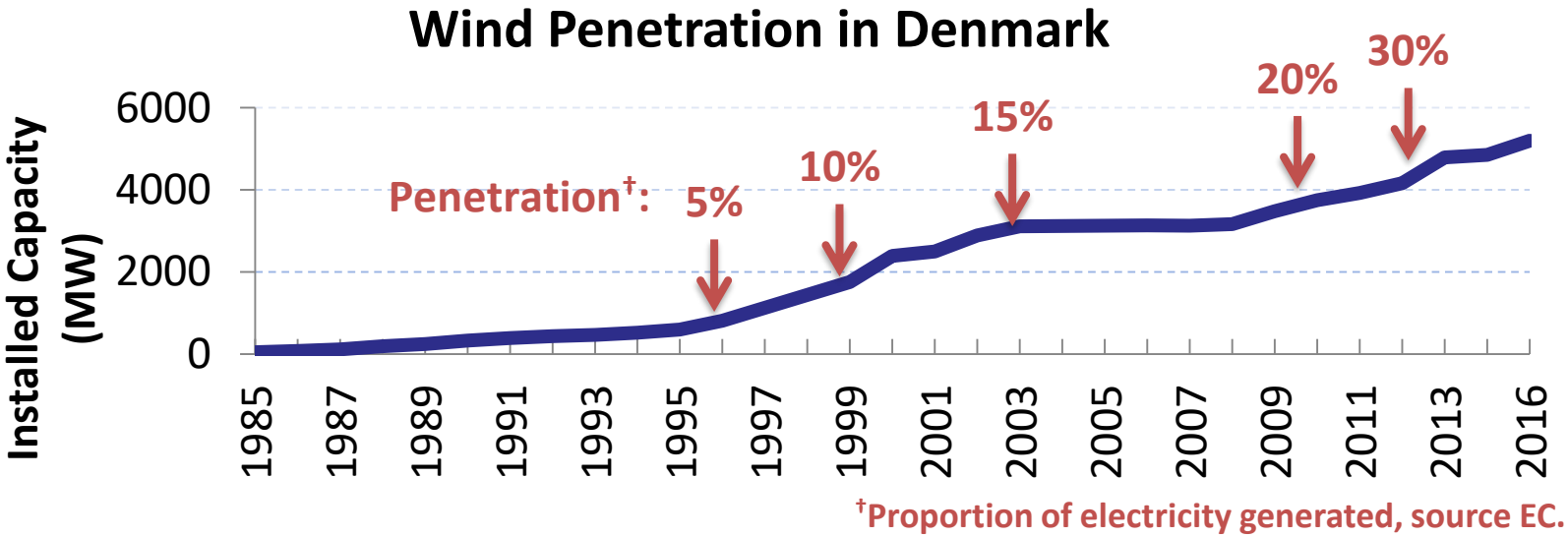


1. When uncertainty is small, deterministic operation is OK
    - Probability distributions well approximated by delta functions
    - Most risks are “low probability, high impact” and treated separately (e.g. generator trip, component failure)

2. When uncertainty is significant, it pays to know what the range of possible futures includes…
    - …and how probable they really are.
    - Many risks of varying severity (much more complex!)

1. Uncertainty Forecasting

What is it and why should I care?



Penetration	Danish Experience
>5%	Basic forecasts are important
>10%	Reliable probabilistic forecasts are needed
>15%	Energy system integration
>20%	Demand side management
>25%	New methods for operating reserves are needed

Source: Henrik Madsen.

1. Uncertainty Forecasting

Forecasting in the Energy Sector

	Very-short-term	Short-term	Medium-Term	Long-term
Timescale	Minutes to Hours	Hours to Days	Weeks to Seasons	Years
Applications/ Users	Balancing/TSO Markets/Traders	Markets/Traders Operational Planning/TSO O&M/Operators	O&M/Operators Markets/Traders Planning/TSO	Planning/ Developers, TSOs, Policy Makers
Methodology	Statistical methods, time series analysis, variations on AR	***Post-processed Numerical Weather Prediction***		Climate Modeling
State-of-the-art/Research Challenges	Large spatial scale (1000s sites), weather regimes, dynamic models	Improving NWP, ***statistical learning*** for post-processing, high-dimensional probabilistic forecasting		Improving and Understanding Climate Models

Things to be forecast now: Demand, Wind, Solar, Price

Things to be forecast in the future: Flexibility, EV charging, DSR, storage (SoC), more prices…

1. Uncertainty Forecasting

Types of Uncertainty Forecast

All forecasts are wrong, but some are useful.

- In many applications there is utility in knowing “how wrong” a [deterministic] forecast could be

- Forecast users are more concerned with the impact on their operation rather than the skill of the forecast itself!

- Two paradigms:
  - The “forecaster’s” perspective – ***error metrics, scoring rules, verification***
  - The “end user’s” perspective – ***value added, decision-support, usability, accountability***

1. Uncertainty Forecasting

Types of Uncertainty Forecast

Quantifying Uncertainty

Level of Detail	Forecaster	Use
Long run performance statistics	Mean Absolute Error etc…	• Heuristics, e.g. *"take action to prepare for error of up to 30%"*
"Simple" Probabilistic Forecasts (intervals, density)	As above, plus reliability, sharpness, etc…	• Situational awareness • Cost-loss decisions • Heuristics e.g. *"take action to prepare for 1-in-10 worst case"*
"Full" Probabilistic Forecasts (multivariate, spatio-temporal trajectories)	As above, plus dependency verification, multivariate energy score etc…	As above, plus: • Stochastic optimisation • Multivariate cost-loss decisions
Extremes	Extreme Value Theory, Ensemble NWP	Low-probability high-cost risk events

1. Uncertainty Forecasting

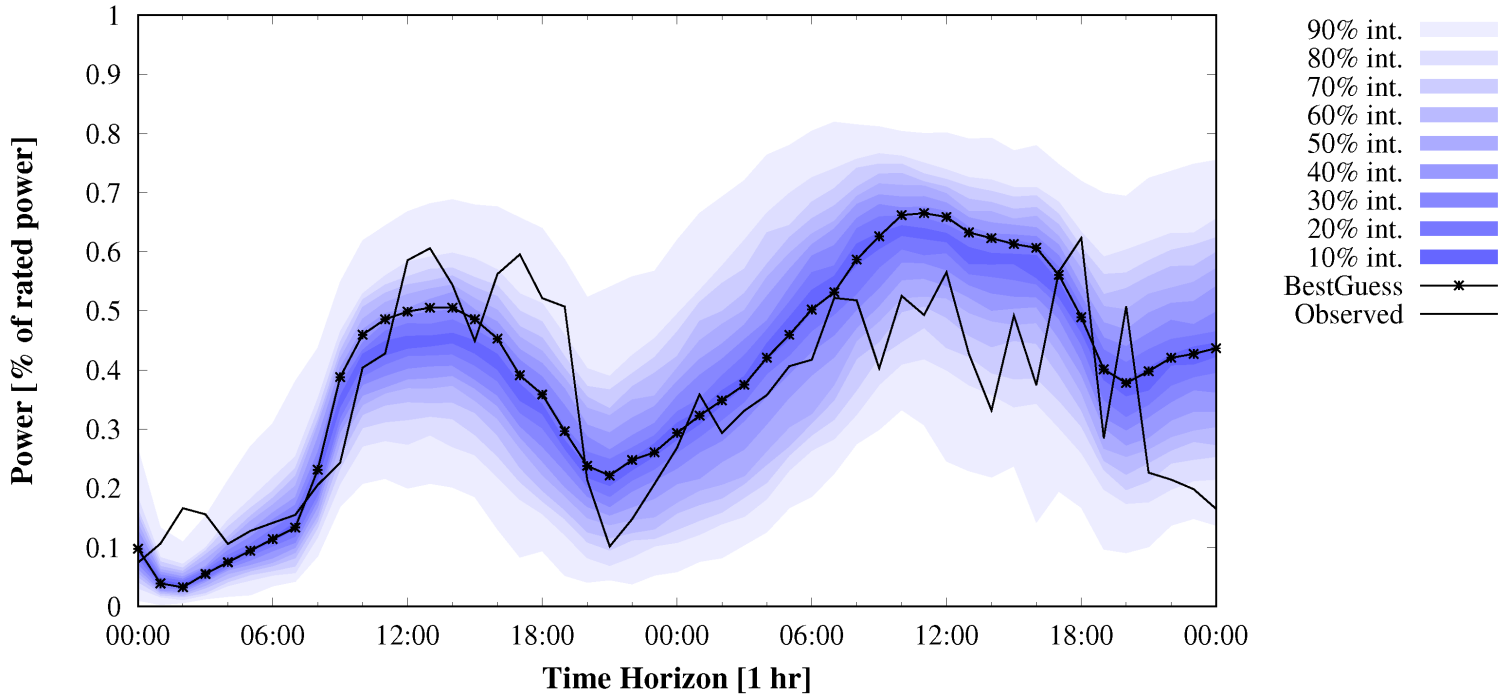
Types of Uncertainty Forecast

Density Forecast

- Prediction at each time point is a probability density function

Interval Forecast

- Fixed probability of observation falling between some upper and lower bound

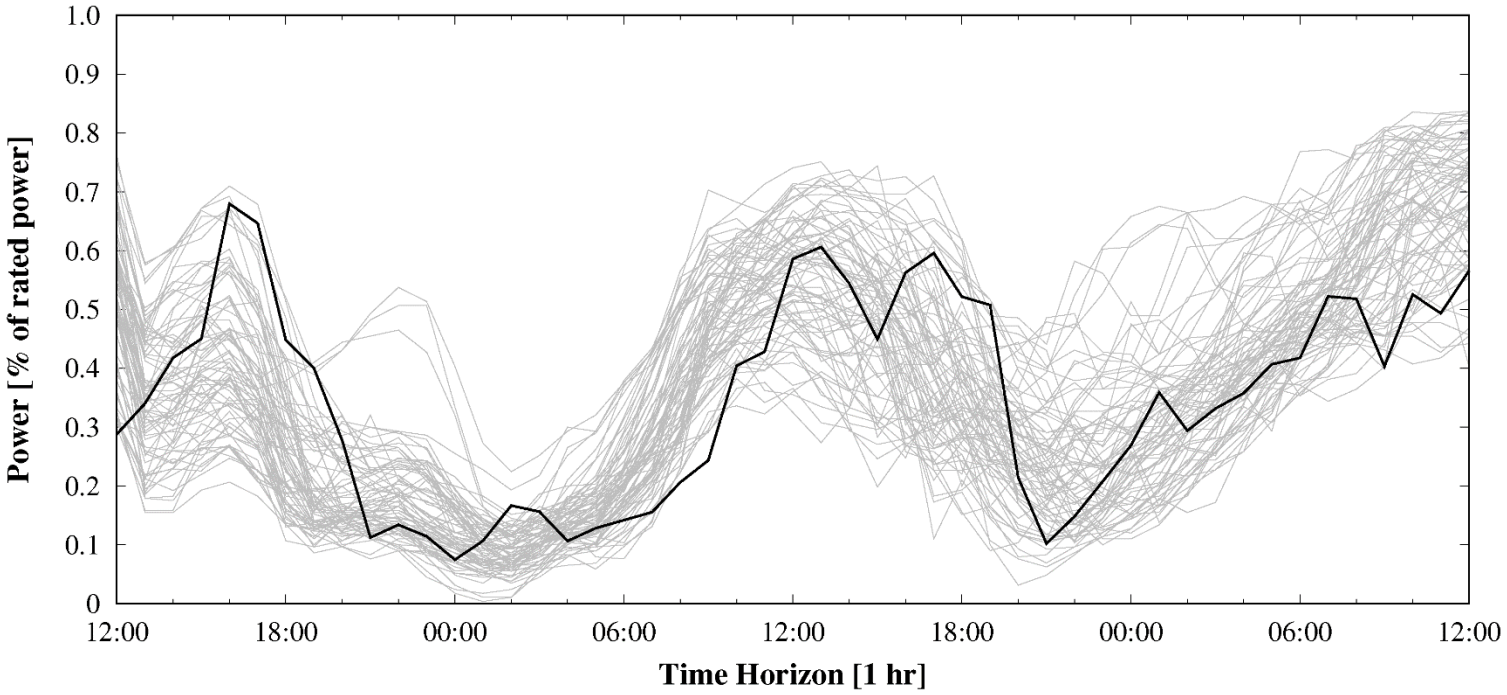


1. Uncertainty Forecasting

Types of Uncertainty Forecast

Scenario Forecast or *Trajectories*

- Set of plausible scenarios
- Samples drawn from multivariate predictive distribution



1. Uncertainty Forecasting

The Cost-loss Model

Decisions making under uncertainty:

- Should we incur cost C to protect against a possible loss L, which has probability p of being realised?

	Adverse Event Occurs	Adverse Event Does Not Occur	Expected Cost
Precautionary Action Taken	C	C	C
Precautionary Action Not Take	L	0	pL

Take action if

$$p > \frac{C}{L}$$

(if you are risk-neutral)

1. Uncertainty Forecasting

The Cost-loss Model

Decisions making under uncertainty:

- Should we invest C for a possible gain L, which has probability p of being realised?

	Positive Event Occurs	Positive Event Does Not Occur	Expected Cost
Investment Action Taken	$C - L$	C	$p(C - L) + (1 - p)C$
Investment Action Not Take	0	0	0

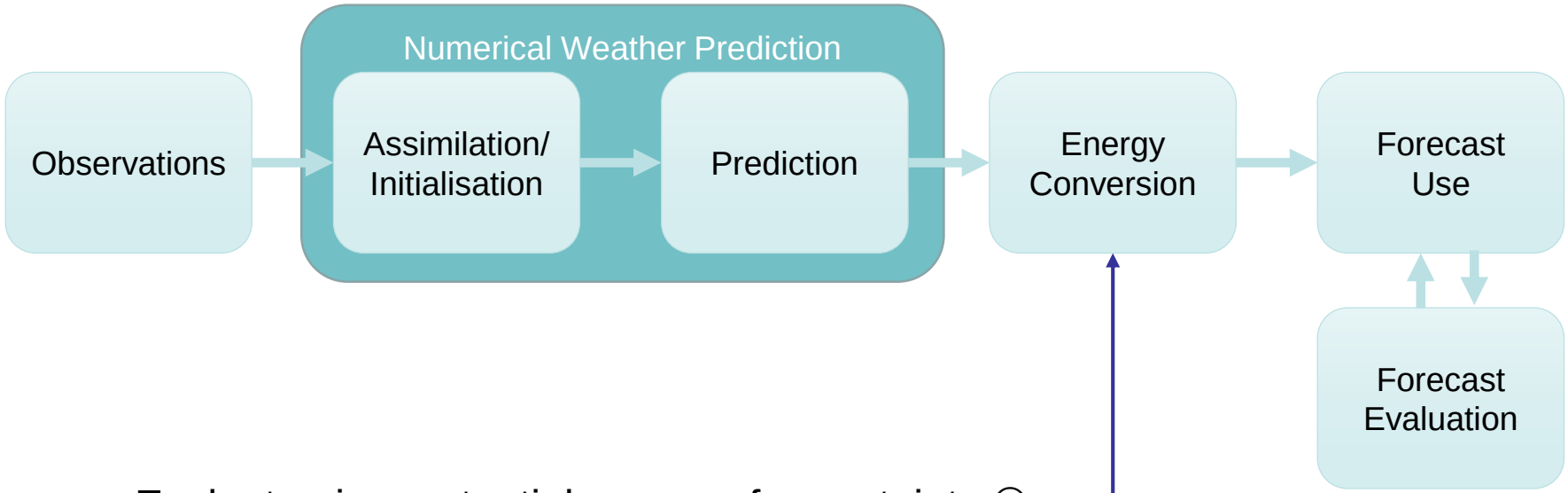
Take action if

$$p > \frac{C}{L}$$

(if you are risk-neutral)

2. Forecasting Model Chain

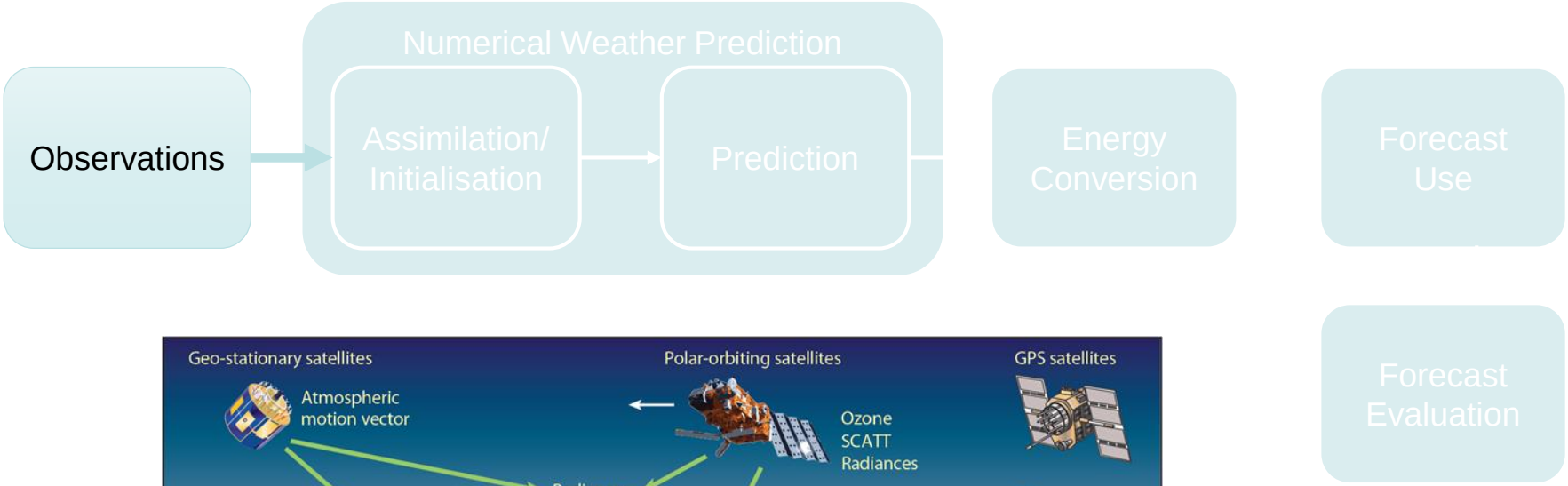
Renewable Energy Forecasting Model Chain

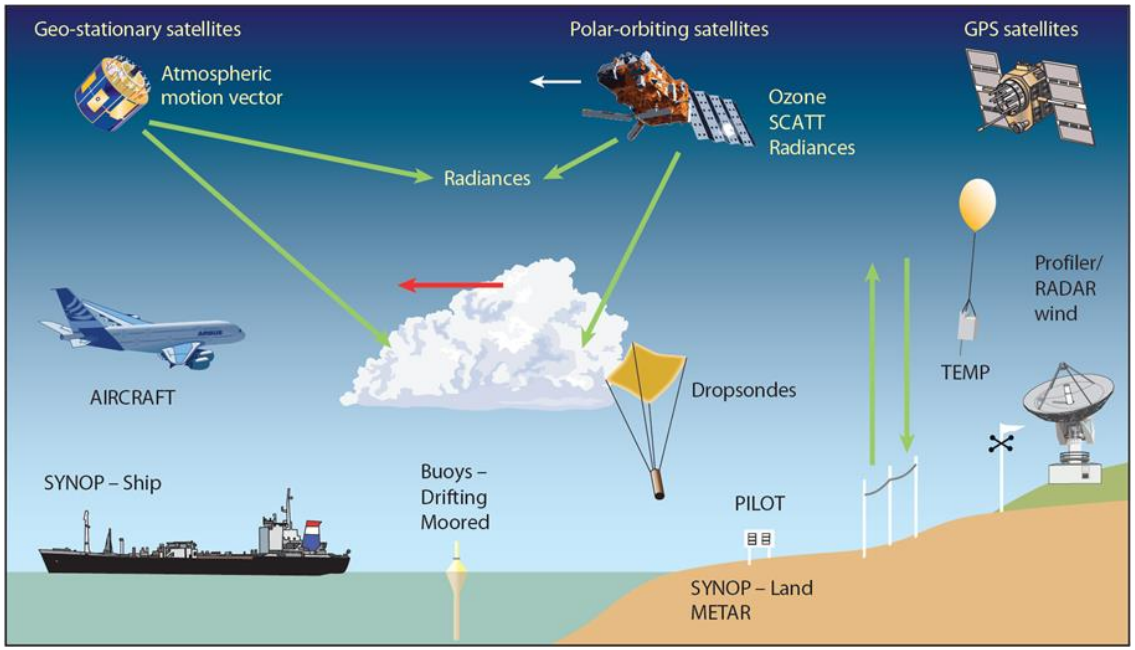


- Each step is a potential source of uncertainty ☹
- By using statistical learning *here* we try to capture the uncertainty in the previous steps to inform decision-making
- Ensemble NWP tries to capture uncertainty at the NPW stage and can also be valuable for energy forecasting

2. Forecasting Model Chain

Renewable Energy Forecasting Model Chain

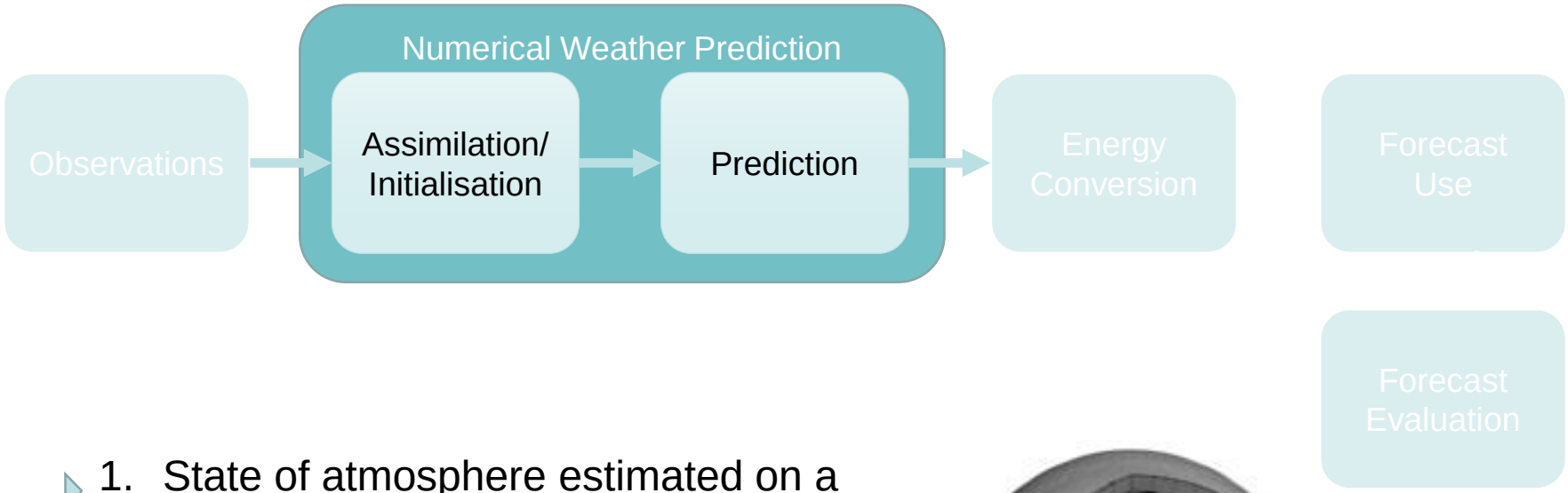




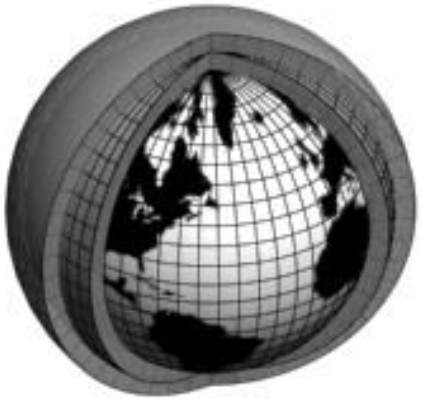
https://www.ecmwf.int/en/research/data-assimilation/observations

2. Forecasting Model Chain

Renewable Energy Forecasting Model Chain

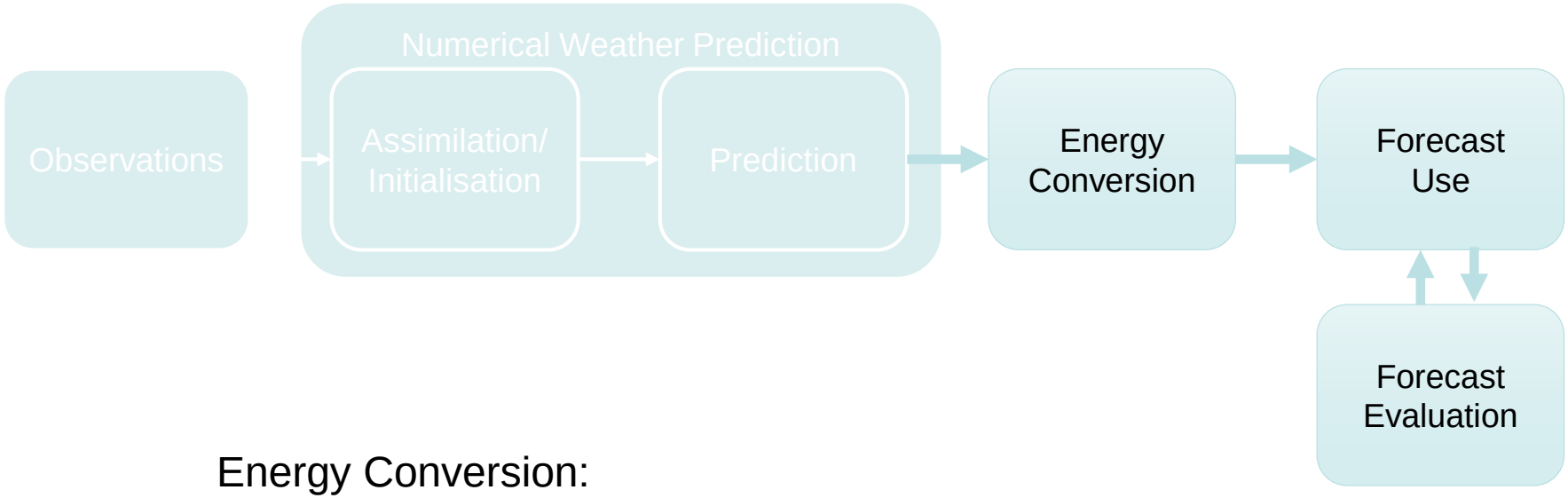


1. State of atmosphere estimated on a grid by “pulling” forecasts towards observations in 4D.
2. Variables at each grid point propagated forwards in time using (linearized) laws of fluid dynamics and other physics
3. (Information exchange between atmospheric and ocean models)



2. Forecasting Model Chain

Renewable Energy Forecasting Model Chain



Energy Conversion:

Statistical Learning!

Forecast Use:

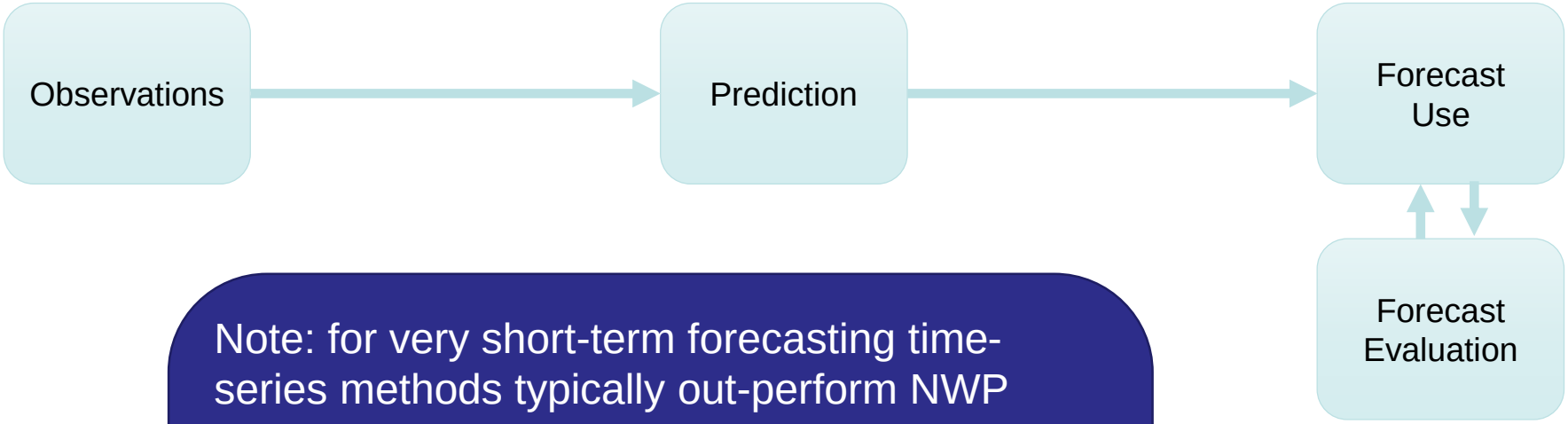
Another Course

Forecast Evaluation:

A little now…

2. Forecasting Model Chain

Aside… Very Short-term Forecasting



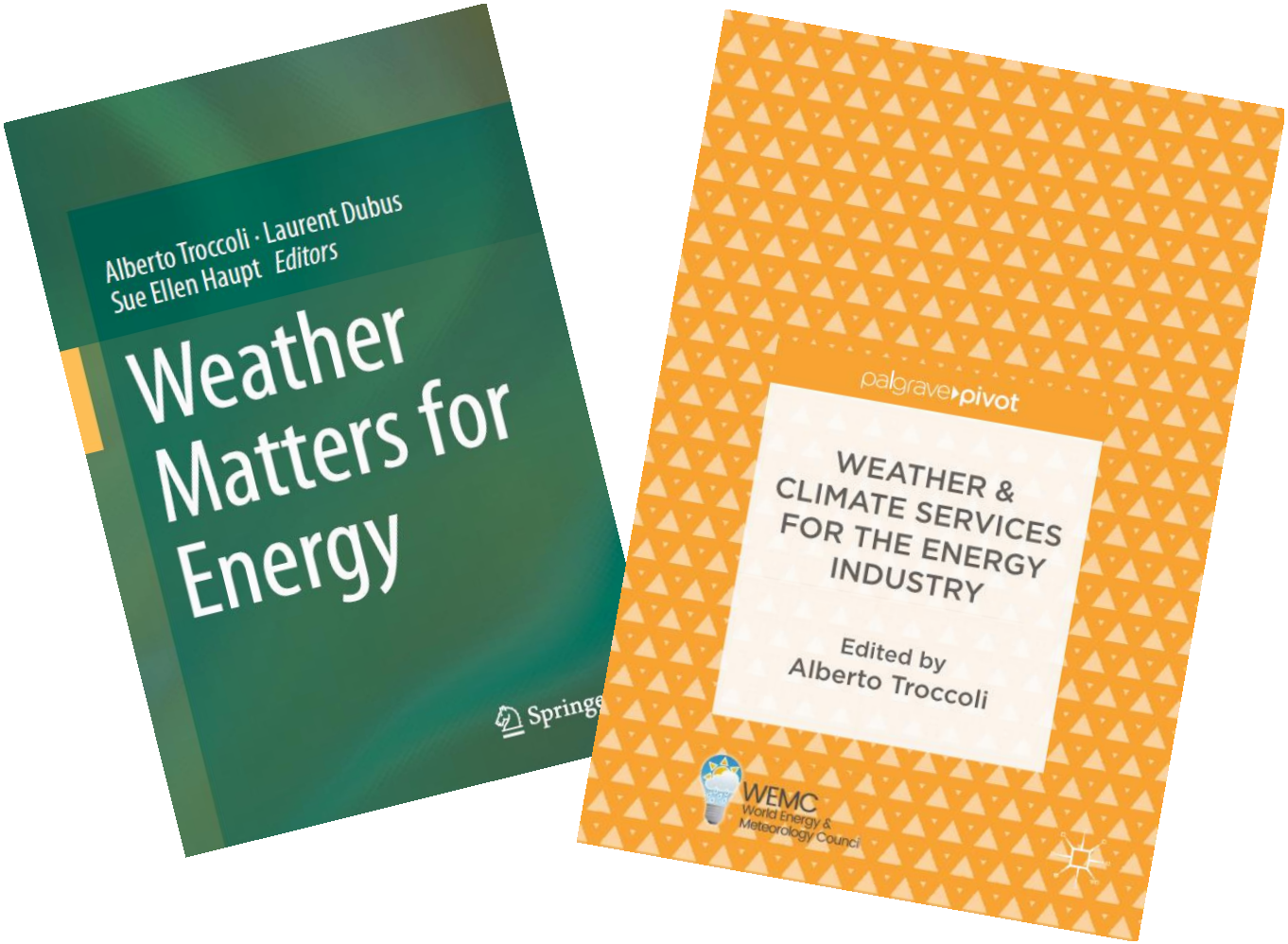
Note: for very short-term forecasting time-series methods typically out-perform NWP

- NWP is already out-of-date when issued
- Often we're interested in high temporal resolution (5,10,15, 30 minutes) in the very-short-term

Many of the statistical learning methods presented today are also useful here.

2. Forecasting Model Chain

Aside…



DOI: 10.1007/978-1-4614-9221-4

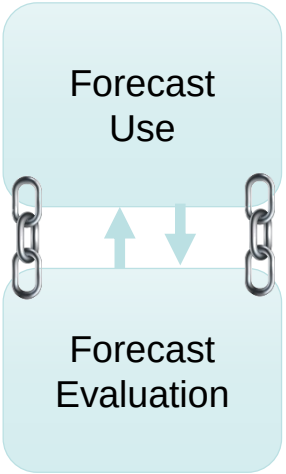
DOI: 10.1007/978-3-319-68418-5

2. Forecasting Model Chain

Forecast Verification

What makes a *'good'* forecast?

- Small average error?
- Low uncertainty/high confidence?
- Reliable uncertainty estimates?
- Better decisions!



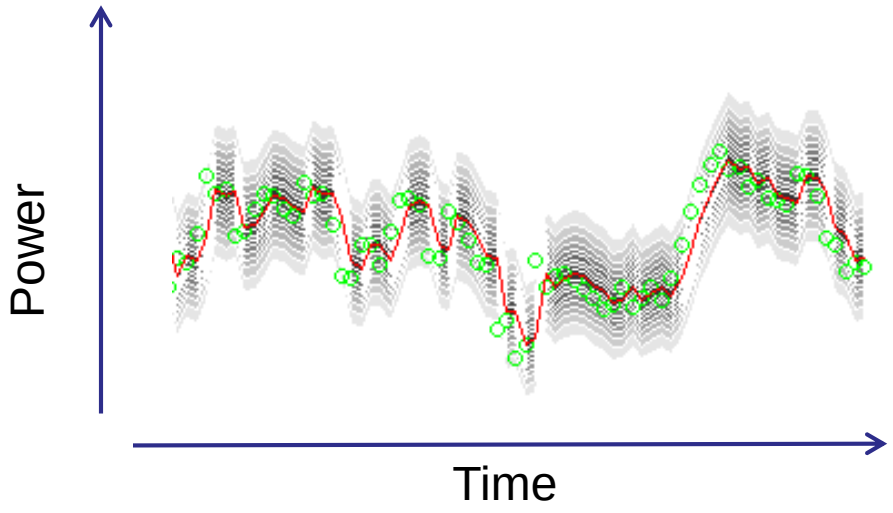
2. Forecasting Model Chain

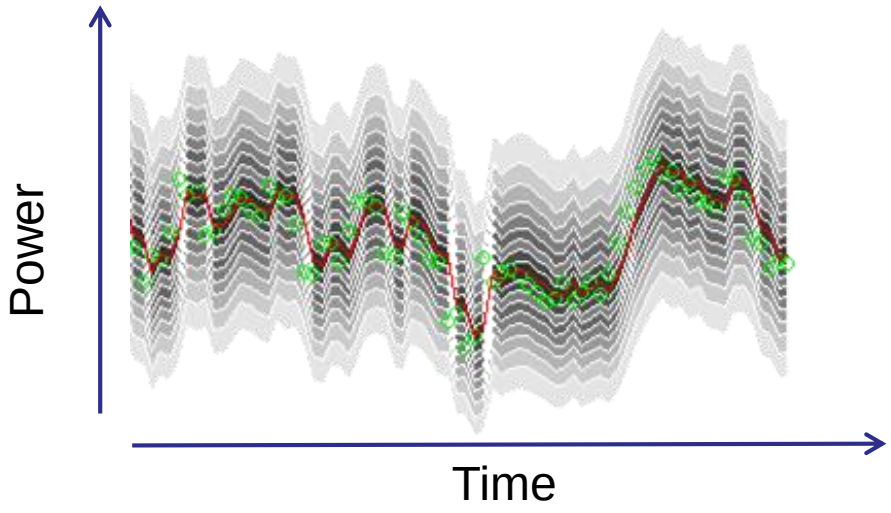
Forecast Verification

Density Forecast

- Sharp (i.e. confident) *subject to calibration/reliability!*
- As we have seen, decision-making is based on specific probability levels – these must be reliable

Sharpness:





2. Forecasting Model Chain

Forecast Verification

Density Forecast

- Sharp (i.e. confident) *subject to calibration/reliability!*
- As we have seen, decision-making is based on specific probability levels – these must be reliable

Calibration/Reliability: Statistical consistency between observations and distributional forecasts.

- E.g. events that are predicted to occur 20% of the time should be observed with a frequency of 20%.
- *Challenge:* For each predictive distribution we produce we only make one observation.

Further reading: Ideas of probabilistic, exceedance, marginal, strong and complete calibration: *Gneiting et al, "Probabilistic forecasts, calibration and sharpness," J. R. Statist. Soc.* B (2007).

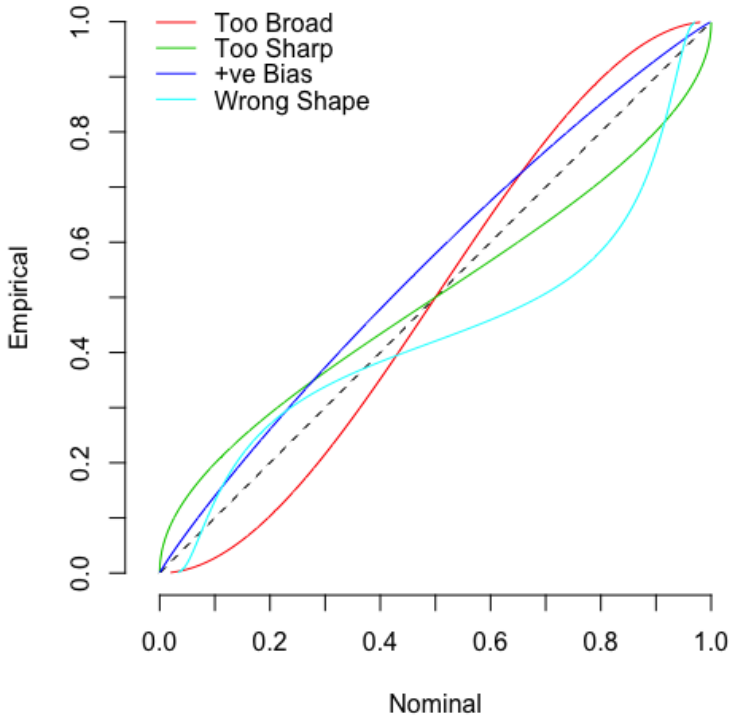
2. Forecasting Model Chain

Forecast Verification

Density Forecast

- Sharp (i.e. confident) *subject to calibration/reliability!*
- As we have seen, decision-making is based on specific probability levels – these must be reliable

Calibration/Reliability:

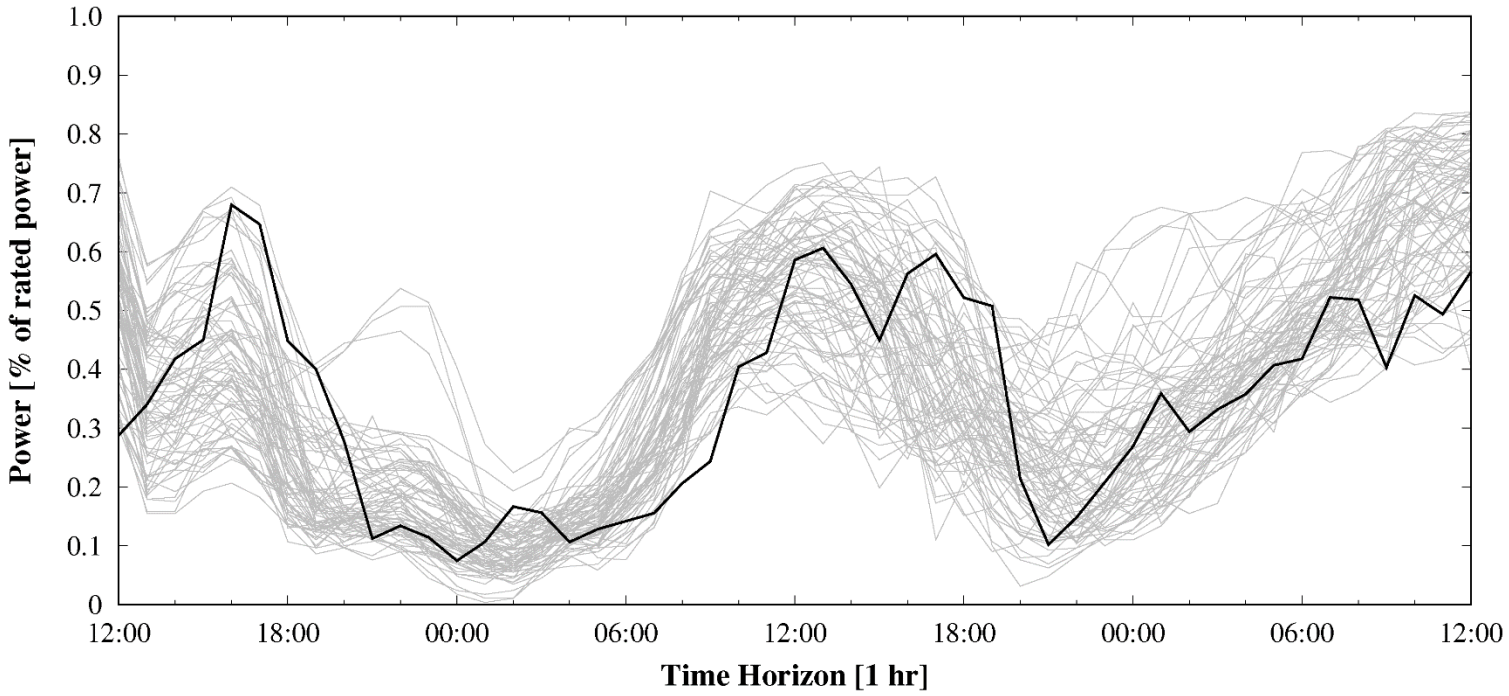


2. Forecasting Model Chain

Forecast Verification

Multi-variate Forecast

- Concept of “reliability” doesn’t generalise to multi-variate case
- Sharp subject to:
    - *Calibration of marginals* (individual variables)
    - *Correct dependency structure* (e.g. spatial, temporal, between variables)



2. Forecasting Model Chain

Forecast Verification

Beware: scoring rules average many individual forecasts…

Forecast Method 1
Very good most of the time, occasionally very bad.

Same Score!!!

Forecast Method 2
Pretty good all of the time.

2. Forecasting Model Chain

Forecast Verification

The Forecaster's Dilemma

- One can successfully predict every extreme event by predicting that it will occur at all times!
- Weighting forecast evaluation by extreme events results in undesirable effects, including the rejection of perfect probabilistic forecasts!
- Important concept: "proper scoring rules" (loosely nothing is better than perfection)

For details see Lerch et al, "Forecaster's Dilemma: Extreme Events and Forecast Evaluation" https://arxiv.org/pdf/1512.09244.pdf

2. Forecasting Model Chain

Forecast Verification for Model Selection

Predictive Power and Cross-validation

K-fold cross-validation makes use of large datasets to assess out-of-sample predictive performance by dividing available data into multiple training and validation sets.

- ✓ Helps avoid overfitting when choosing values of hyper-parameters
- ✓ Give indication of final performance
- ➢ Requires sufficient data that each training set is representative
- ➢ Computationally demanding

Train	Train	Train	Validation
Train	Train	Validation	Train
Train	Validation	Train	Train
Validation	Train	Train	Train

3. Linear Regression and Extensions

Contents

What I will cover in this section:

- Parametric Uncertainty Forecasting
    - Linear models
    - Generalised Additive Models (for Location, Scale and Shape…)
- Regularisation
    - Ridge
    - LASSO
    - Splines
- Non-parametric Uncertainty Forecasting
    - Quantile Regression

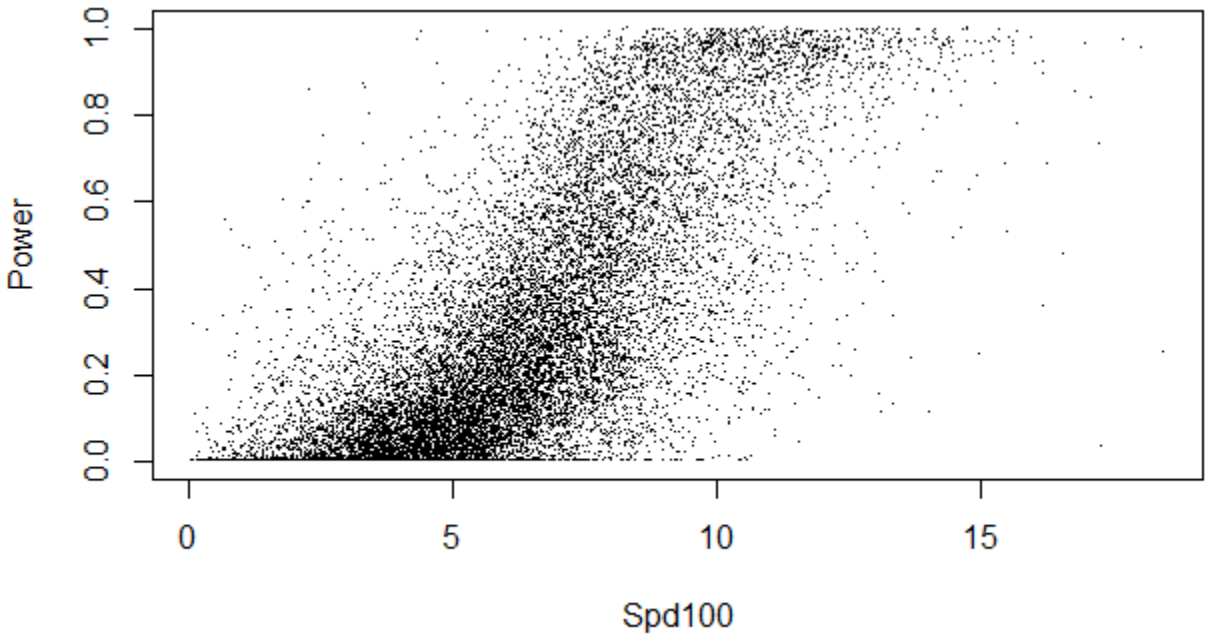
3. Linear Regression and Extensions

Contents

Illustrative Examples Based on GEFcom2014 Wind Power Data

Caveats:

- Only wind speed included as input for simple visualisation
- Important things such as data cleaning, parameter tuning, cross-validation have *not* been given much consideration
- R code available with slides



3. Linear Regression and Extensions

Basic Formulation of Supervised Learning

$$y_t = f(x_{1,t}, x_{2,t}, \dots) + \epsilon_t$$

Find $f(\cdot)$ to minimise some function of $\epsilon_t, \mathrm{t} = 1, \dots, \mathrm{T}$

…or describe the statistics of ϵ_t to quantify uncertainty.

3. Linear Regression and Extensions

Loss Functions

Mean Squared Error:

$$L(\epsilon_1, \dots, \epsilon_T) = \frac{1}{T}\sum_{i=1}^{T} \epsilon_i^2$$

Mean Absolute Error:

$$L(\epsilon_1, \dots, \epsilon_T) = \frac{1}{T}\sum_{i=1}^{T} |\epsilon_i|$$

Quantile Loss (Pinball) for τ^{th} quantile:

$$L_\tau(\epsilon_1, \dots, \epsilon_T) = \frac{1}{T}\sum_{\epsilon_i \geq 0} \tau\epsilon_i + \frac{1}{T}\sum_{\epsilon_i < 0} (\tau - 1)\epsilon_i$$

3. Linear Regression and Extensions

Linear Models

Suppose $f\left(x_{1,t}, x_{2,t}, \ldots\right)$ is a linear combination of $x_{1,t}, x_{2,t}, \ldots$

$$y_t = f\left(x_{1,t}, x_{2,t}, \ldots\right) + \epsilon_t$$

$$y_t = \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \epsilon_t$$

$$y_t = \boldsymbol{x}_t^{\top}\boldsymbol{\beta} + \epsilon_t$$

3. Linear Regression and Extensions

Linear Models

Parameters $\boldsymbol{\beta}$ to be estimated, or *learnt*, to minimise chosen loss function over some set of examples.

$$\begin{bmatrix} y_1 \\ \vdots \\ y_T \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_1^\top \\ \vdots \\ \boldsymbol{x}_T^\top \end{bmatrix} \boldsymbol{\beta} + \begin{bmatrix} \epsilon_1 \\ \vdots \\ \epsilon_T \end{bmatrix}$$

$$\mathrm{Y} = X^\top \boldsymbol{\beta} + \boldsymbol{\epsilon}$$

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, L(Y - X^\top \boldsymbol{\beta})$$

3. Linear Regression and Extensions

Linear Models

Deterministic Forecasting: Ordinary Least Squares

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} (Y - X^{\top}\boldsymbol{\beta})^{\top}(Y - X^{\top}\boldsymbol{\beta})$$

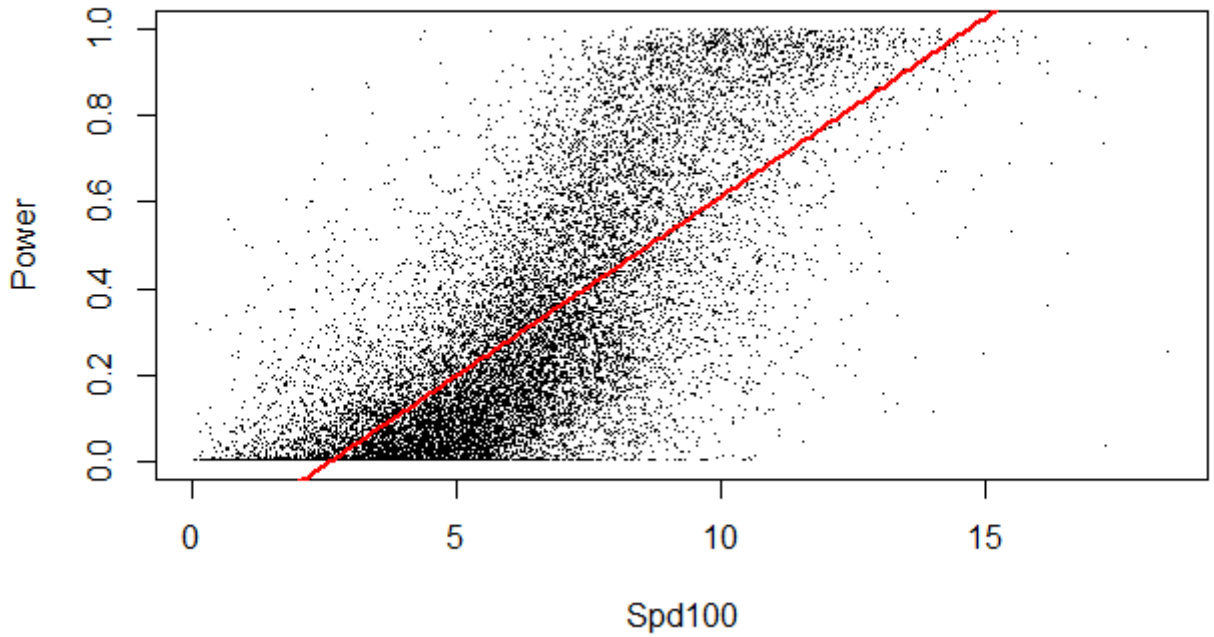
$$\frac{d}{d\boldsymbol{\beta}}(Y - X^{\top}\boldsymbol{\beta})^{\top}(Y - X^{\top}\boldsymbol{\beta}) = 0$$

$$\widehat{\boldsymbol{\beta}} = (X^{\top}X)^{-1}X^{\top}Y$$

3. Linear Regression and Extensions

Linear Models

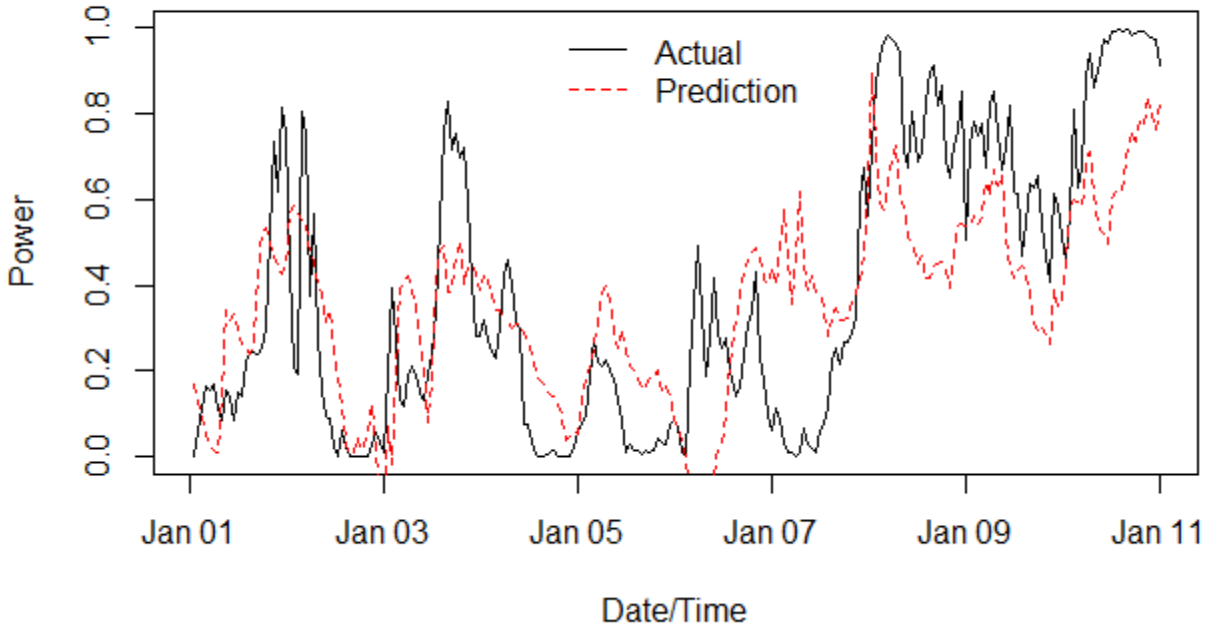
Deterministic Forecasting: Ordinary Least Squares



3. Linear Regression and Extensions

Linear Models

Deterministic Forecasting: Ordinary Least Squares



3. Linear Regression and Extensions

Linear Models

Uncertainty Forecasting: Maximum Likelihood Estimation

$$\mathrm{Y} = X^{\top}\boldsymbol{\beta} + \boldsymbol{\epsilon}$$

If ϵ_t follow some distribution, find the parameters $\boldsymbol{\beta}$ that maximises the likelihood of observing (Y, X)

3. Linear Regression and Extensions

Linear Models

Uncertainty Forecasting: Maximum Likelihood Estimation

$$\epsilon_i \sim N(0, \sigma) \quad \forall i$$

$$\mathrm{Y} = X^\top \boldsymbol{\beta} + \boldsymbol{\epsilon}$$

$$\mathrm{P}(\mathrm{Y}, \mathrm{X}; \boldsymbol{\beta}) = \prod_i \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{\epsilon_i^2}{2\sigma^2}}$$

Called “log-likelihood” ℓ

$$\mathrm{LogP}(\mathrm{Y}, \mathrm{X}; \boldsymbol{\beta}) \propto \sum_i -\frac{\epsilon_i^2}{2\sigma^2}$$

Look familiar?

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\mathrm{argmin}} (Y - X^\top \boldsymbol{\beta})^\top (Y - X^\top \boldsymbol{\beta})$$

3. Linear Regression and Extensions

Linear Models

Uncertainty Forecasting: Maximum Likelihood Estimation

$$\epsilon_i \sim N(0, \sigma) \quad \forall i$$

$$\widehat{\boldsymbol{\beta}} = (X^\top X)^{-1} X^\top Y$$

Standard deviation σ is the sample s.d. of ϵ_i using $\widehat{\boldsymbol{\beta}}$.

3. Linear Regression and Extensions

Linear Models

Uncertainty Forecasting: Maximum Likelihood Estimation

$$\epsilon_i \sim N(0, \sigma) \quad \forall i$$

We can now write down a “predictive distribution” for y_t

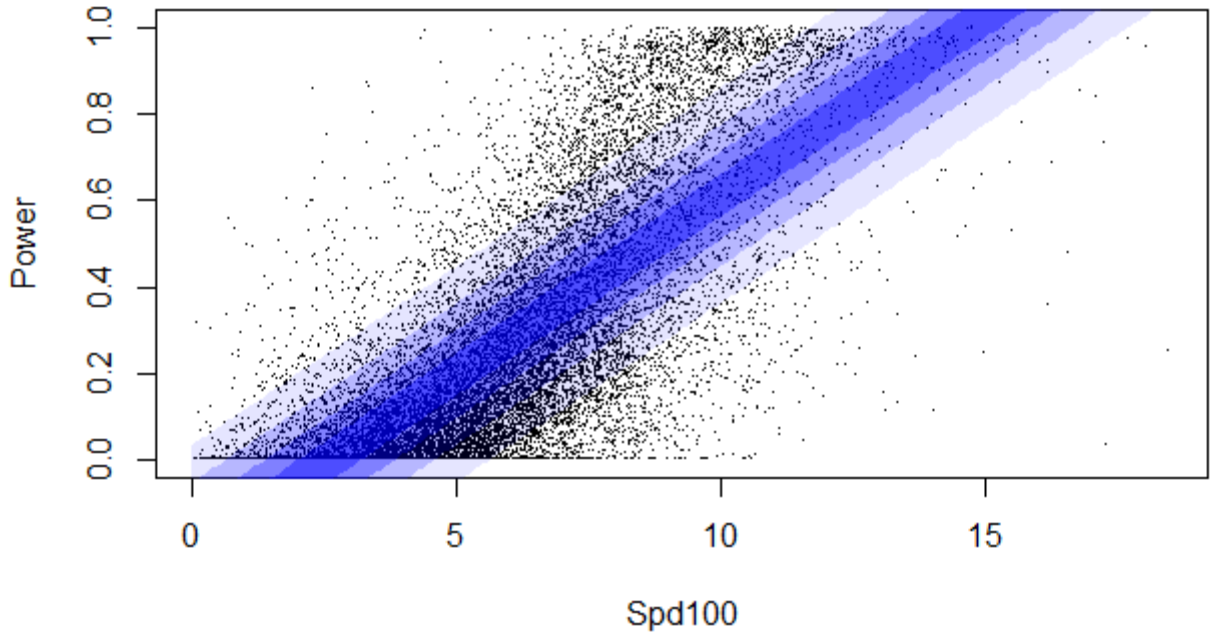
$$y_t = \boldsymbol{x}_t^\top \boldsymbol{\beta} + \epsilon_t$$

$$y_t \sim N(\boldsymbol{x}_t^\top \boldsymbol{\beta}, \sigma)$$

3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

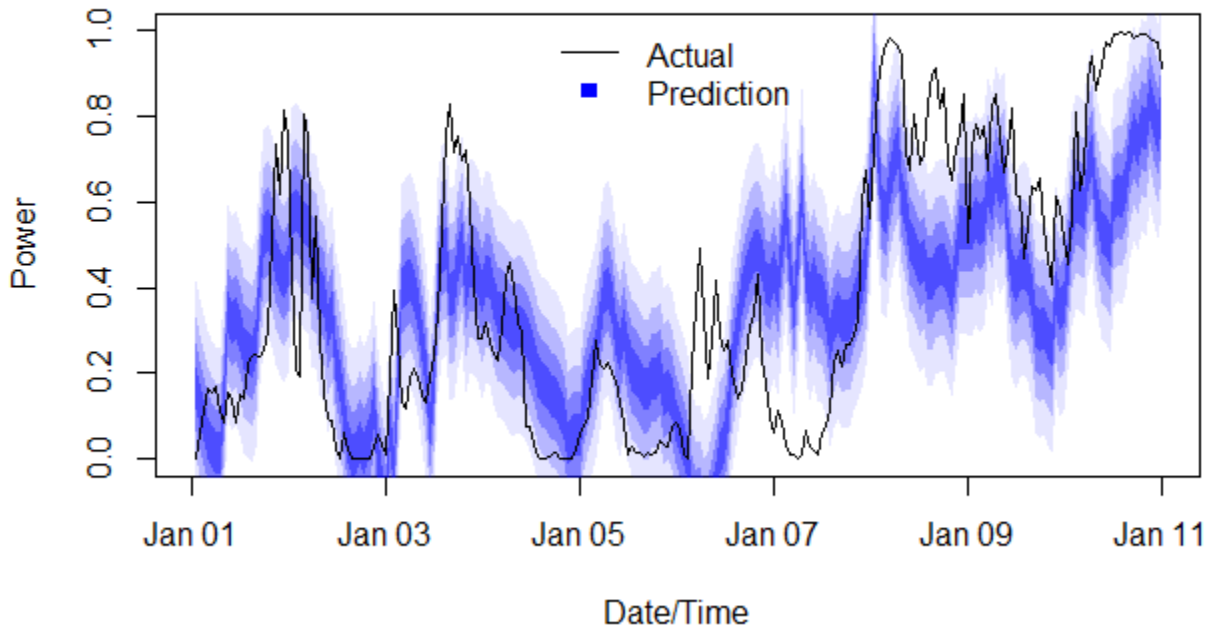
Example: MLE of Gaussian



3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

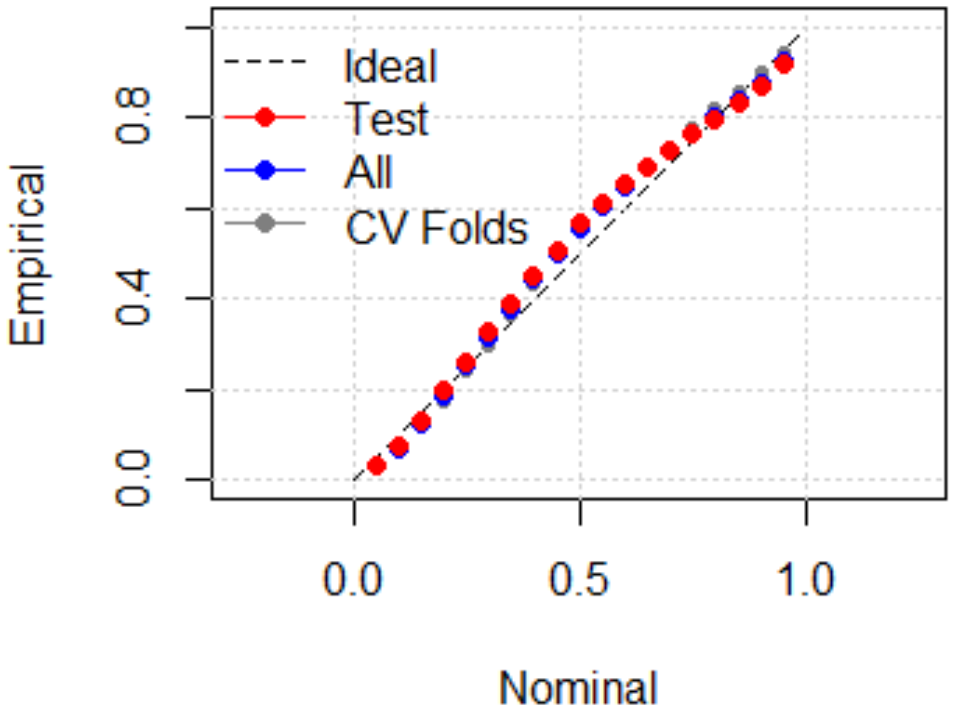
Example: MLE of Gaussian



3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: MLE of Gaussian



3. Linear Regression and Extensions

Linear Models

Uncertainty Forecasting: Maximum Likelihood Estimation

What if ϵ isn't Gaussian?

Option 1: Transform your data or increase complexity of your model

Option 2: MLE for another parametric distribution

Option 3: Non-parametric

3. Linear Regression and Extensions

Generalised Linear Models

Uncertainty Forecasting with Generalised Linear Models
(a fancy name for something quite simple)

$$g(y_t) = \boldsymbol{x}_t^\top \boldsymbol{\beta} + \epsilon_t$$

- Most results for linear models still hold
- $g(\cdot)$ called the “link function”
- Common transformations:
    - Log (for positive *spiky* data, e.g. volatile prices)
    - Probit/Logistic functions (transform $(0,1)$ to$(-\infty, \infty)$)
- Sometime called “variance stabilisation”

3. Linear Regression and Extensions

Generalised Additive Models

Generalised Additive Models

$$g(y_t) = \boldsymbol{x}_t^{\top}\boldsymbol{\beta} + f_1(x_{1,t}) + f_2(x_{2,t}) + \cdots + \epsilon_t$$

We can use results for linear models if $f(x)$ takes the form of a linear model…

$$f(x) = \sum_{i=1}^{q} b_i(x)\beta_i$$

3. Linear Regression and Extensions

Generalised Additive Models

Generalised Additive Models
Basis Functions

$$g(y_t) = \boldsymbol{x}_t^{\top}\boldsymbol{\beta} + f_1(x_{1,t}) + f_2(x_{2,t}) + \cdots + \epsilon_t$$

$b_i(\cdot)$ can be chosen freely! If you know something about your data chose some relevant functions…

$$f(x) = \sum_{i=1}^{q} b_i(x)\beta_i$$

3. Linear Regression and Extensions

Generalised Additive Models

Generalised Additive Models
Some Useful Basis Functions (for information only)

$$f(x) = \sum_{i=1}^{q} b_i(x)\beta_i$$

Polynomial Regression

$$f(x) = \sum_{i=1}^{q} x^i \beta_i$$

Local Regression (LOESS)

$$f(x) = \sum_{j} \sum_{i=1}^{q} w_j(x) x^i \beta_i$$

$w_j(x)$: weight function, tri-cube is popular, others possible.

3. Linear Regression and Extensions

Generalised Additive Models

Generalised Additive Models
Basis Functions

$$f(x) = \sum_{i=1}^{q} b_i(x)\beta_i$$

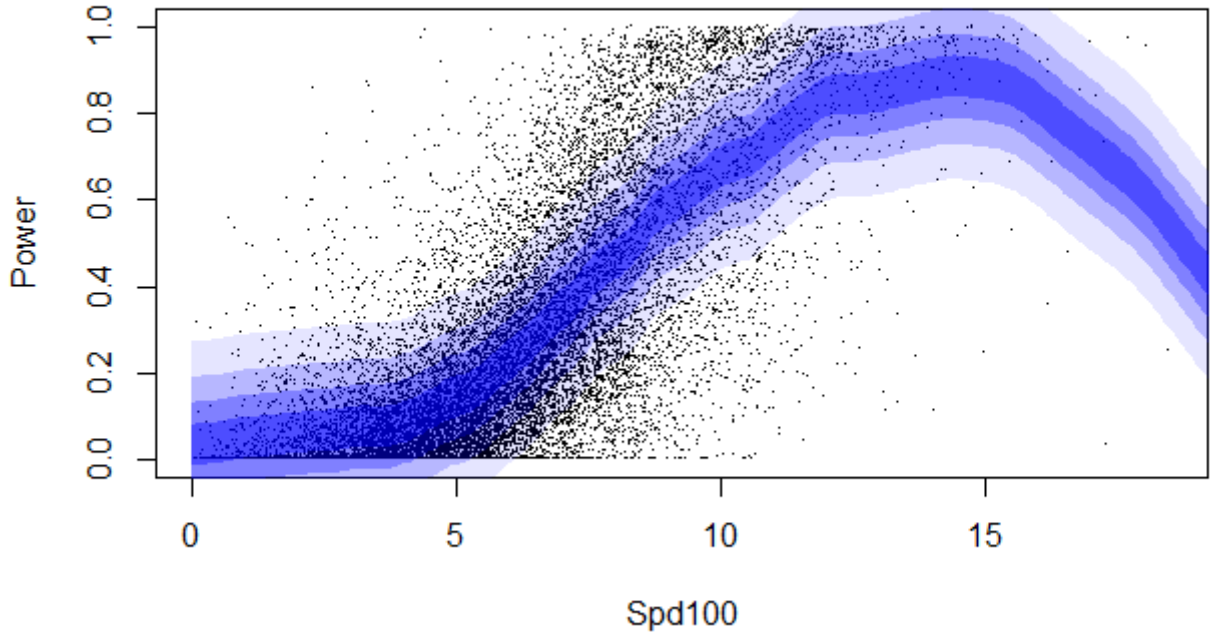
Spline Basis

- Cubic Splines: sections of cubic polynomials joined at “knots”. First and second derivative continuous at knots.
- B-splines: each basis function only non-zero in locality of it’s knot.
- P-splines: penalised B-splines, very flexible.

3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: Cubic Splines



3. Linear Regression and Extensions

Regularisation

Increasing complexity robustly…

Possible Improvement	Risk
Expand set of explanatory variables (new data or “engineered” features)	Poor parameter estimates, especially for correlated features.
Increase complexity of splines/basis functions	Over-fitting

Mitigation: *Regularisation*

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left[\sum_t L(y_t, \boldsymbol{x}_t\boldsymbol{\beta}) + J(\boldsymbol{\beta}) \right]$$

3. Linear Regression and Extensions

Regularisation

Possible Improvement	Risk
Expand set of explanatory variables (new data or “engineered” features)	Poor parameter estimates, especially for correlated features.

- Correlated features means correlated columns of X and the covariance matrix $R = X^{\mathsf{T}}X$
- This leads of the determinant of $X^{\mathsf{T}}X$ being very small and the OLS solution is “ill-conditioned”

$$\widehat{\boldsymbol{\beta}} = (X^{\mathsf{T}}X)^{-1}X^{\mathsf{T}}Y$$

Gets very big…

Gets very small…

3. Linear Regression and Extensions

Regularisation

Penalised Least Squares: Ridge

Ridge Regression:

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}[|Y - X^{\top}\boldsymbol{\beta}|_2^2 + \lambda|\boldsymbol{\beta}|_2^2]$$

Closed form solution:

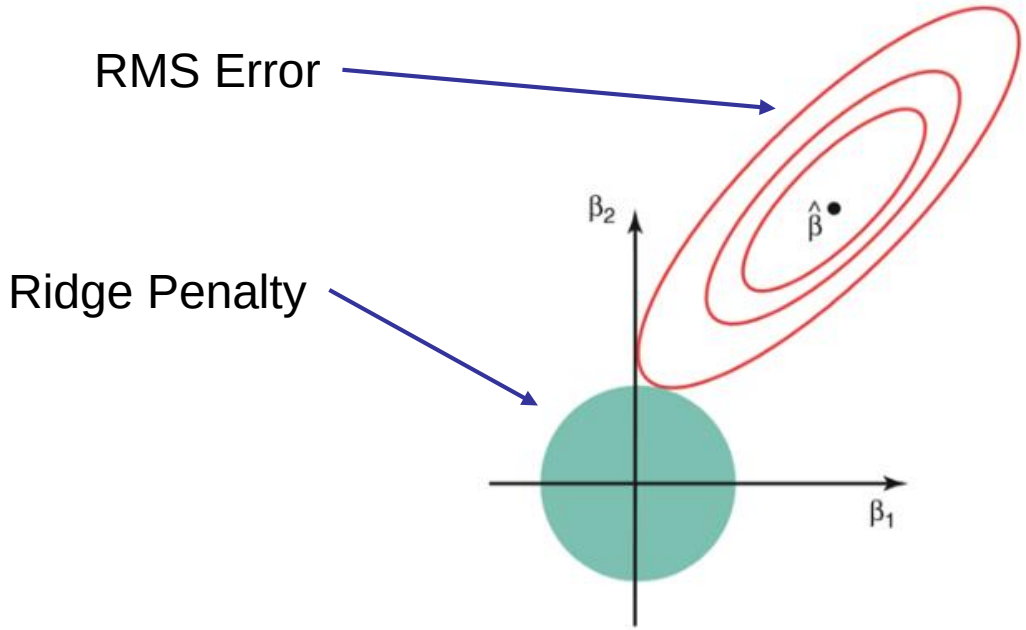
$$\widehat{\boldsymbol{\beta}} = (X^{\top}X + \lambda I)^{-1}X^{\top}Y$$

Hyper-parameter λ typically chosen via cross-validation

3. Linear Regression and Extensions

Regularisation

Penalised Least Squares: Ridge



Credit : An Introduction to Statistical Learning by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani

3. Linear Regression and Extensions

Regularisation

Penalised Least Squares: LASSO

Least Absolute Shrinkage and Selection Operator:

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left[|Y - X^{\top}\boldsymbol{\beta}|_2^2 + \lambda|\boldsymbol{\beta}|_1\right]$$

No closed form solution, must be solved numerically.
Thankfully, some neat tricks make this pretty efficient.

Hyper-parameter λ typically chosen via cross-validation.

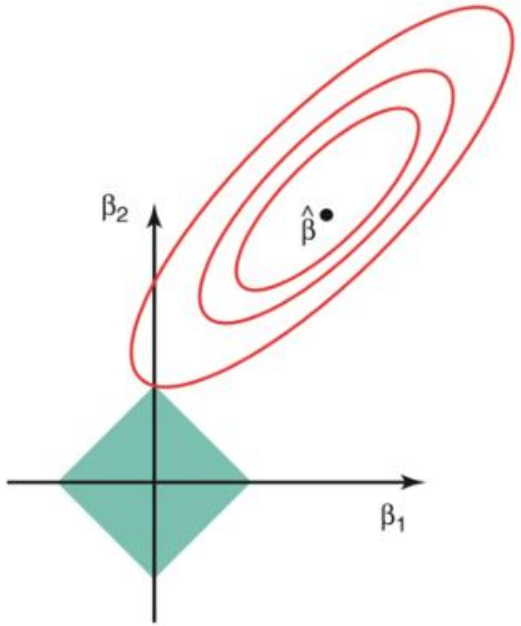
3. Linear Regression and Extensions

Regularisation

Penalised Least Squares: LASSO

Least Absolute Shrinkage and **Selection** Operator:

- Some parameters driven to <u>exactly zero</u>
- LASSO performs parameter selection and estimation simultaneously.

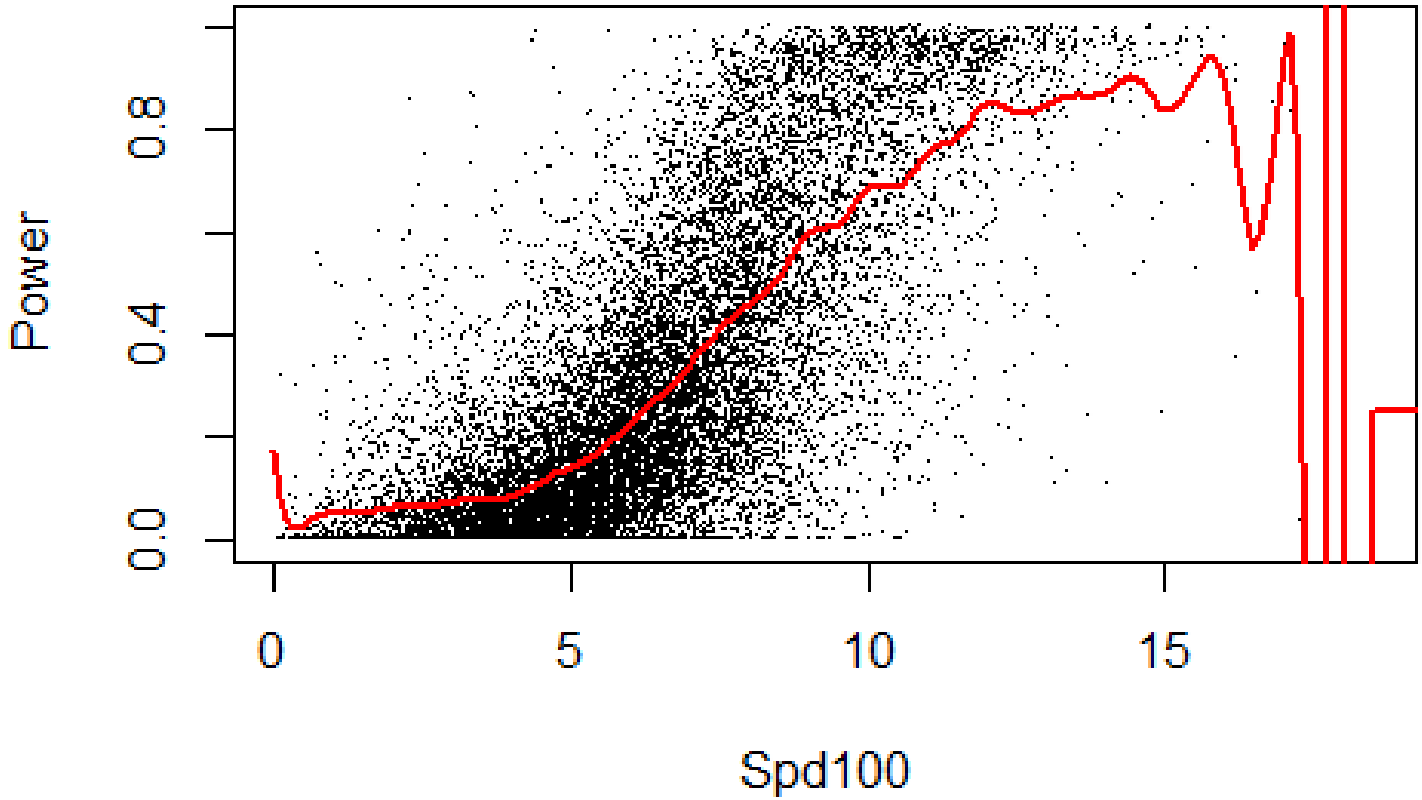


Credit : An Introduction to Statistical Learning by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani

3. Linear Regression and Extensions

Regularisation

Possible Improvement	Risk
Increase complexity of splines/basis functions	Over-fitting



3. Linear Regression and Extensions

Regularisation

Penalised Regression Splines

- Penalise the second derivative, or “wiggliness” of the spline:

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left[|Y - X^{\top}\boldsymbol{\beta}|_2^2 + \int \; [f''(x)]^2 dx \right]$$

- Because splines are of the form $f(x) = \sum_{i=1}^{q} b_i(x)\beta_i$ this becomes:

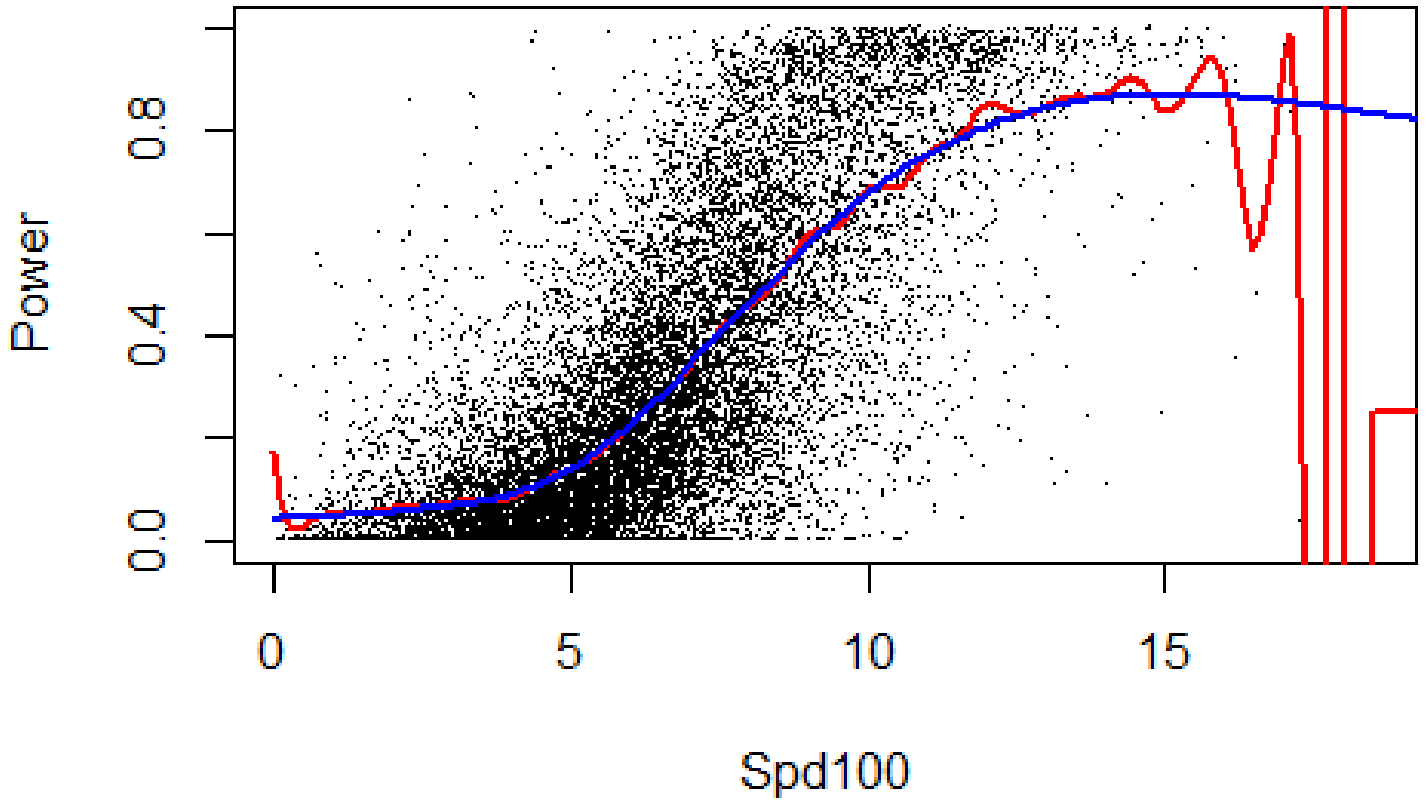
$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} [|Y - X^{\top}\boldsymbol{\beta}|_2^2 + \boldsymbol{\beta}^{\top} S \boldsymbol{\beta}]$$

- The matrix S is known provided the basis functions are twice differentiable

3. Linear Regression and Extensions

Regularisation

Possible Improvement	Risk
Increase complexity of splines/basis functions	Over-fitting



3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Option 2
Other Parametric Distributions

- In the real world, high order moments (variance, skewness, kurtosis) are not fixed and should be treated as dependent variables.
- Closed-form solutions for MLEs of many multiple-parameter distributions do not exist

3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Option 2
Other Parametric Distributions

- So far we've considered GAMs for the mean or *location* parameter of the Gaussian distribution:

$$g_1(\boldsymbol{\mu}) = \boldsymbol{X}_1\boldsymbol{\beta}_1$$

- We can expand this to other parameters, *scale* and *shape,* too:

$$g_2(\boldsymbol{\sigma}) = \boldsymbol{X}_2\boldsymbol{\beta}_2$$

$$g_3(\boldsymbol{\nu}) = \boldsymbol{X}_3\boldsymbol{\beta}_3$$

$$g_4(\boldsymbol{\tau}) = \boldsymbol{X}_4\boldsymbol{\beta}_4$$

3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Option 2
Other Parametric Distributions

- Estimation of $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3, \boldsymbol{\beta}_4$: numerical methods

Rigby-Stasinopoulos

- Calculate partial derivatives of likelihood function
- Sequentially estimate each $\boldsymbol{\beta}_i$ using recent estimate of others until convergence

Cole-Green

- Calculate partial derivatives of likelihood function
- Calculate cross derivatives of likelihood function
- Update step in direction of steepest decent

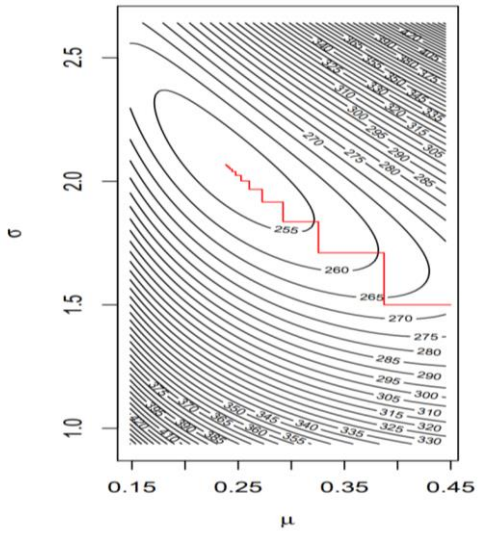
3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

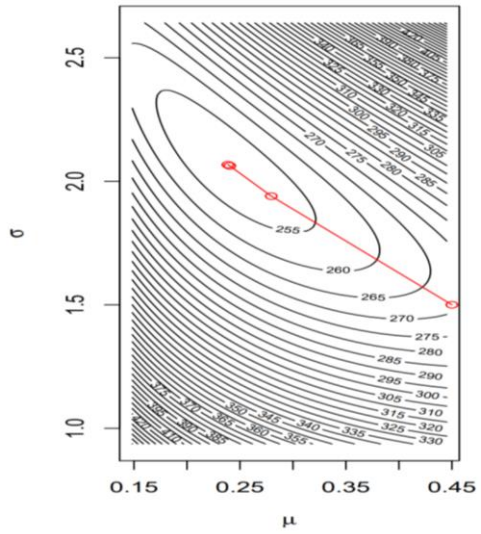
Option 2
Other Parametric Distributions

- Estimation of $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3, \boldsymbol{\beta}_4$: numerical methods

Rigby-Stasinopoulos



Cole-Green



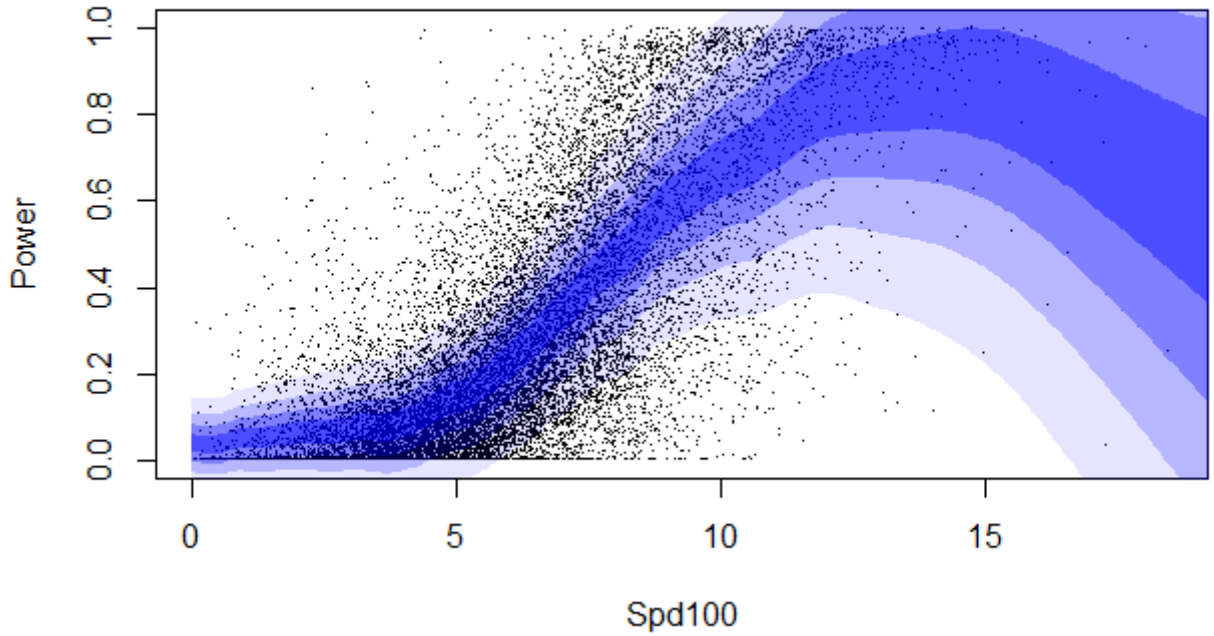
"Flexible Regression and Smoothing The GAMLSS packages in R," Mikis Stasinopoulos, Bob Rigby, Vlasios Voudouris, Gillian Heller and Fernanda De Bastiani, www.gamlss.com

3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: GAMLSS

$$y_t \sim N(\mu = \boldsymbol{x}_{1,t}^{\mathrm{CS}}\boldsymbol{\beta}_1, \sigma = e^{\boldsymbol{x}_{2,t}\boldsymbol{\beta}_2})$$

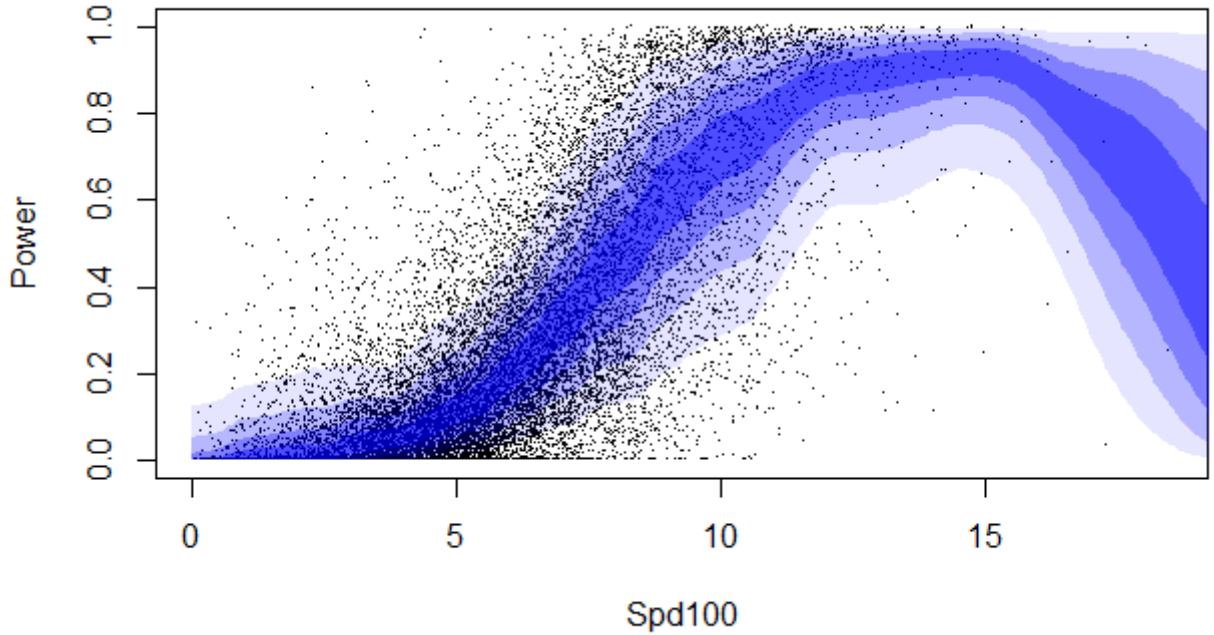


3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: GAMLSS

$$y_t \sim InflatedBeta(\boldsymbol{x}_t; a, b, p_1, p_2) = \begin{cases} p_1 & x = 0 \\ (1 - p_1 - p_2) \times Beta(\boldsymbol{x}_t; a, b) & 0 < x < 1 \\ p_2 & x = 1 \end{cases}$$

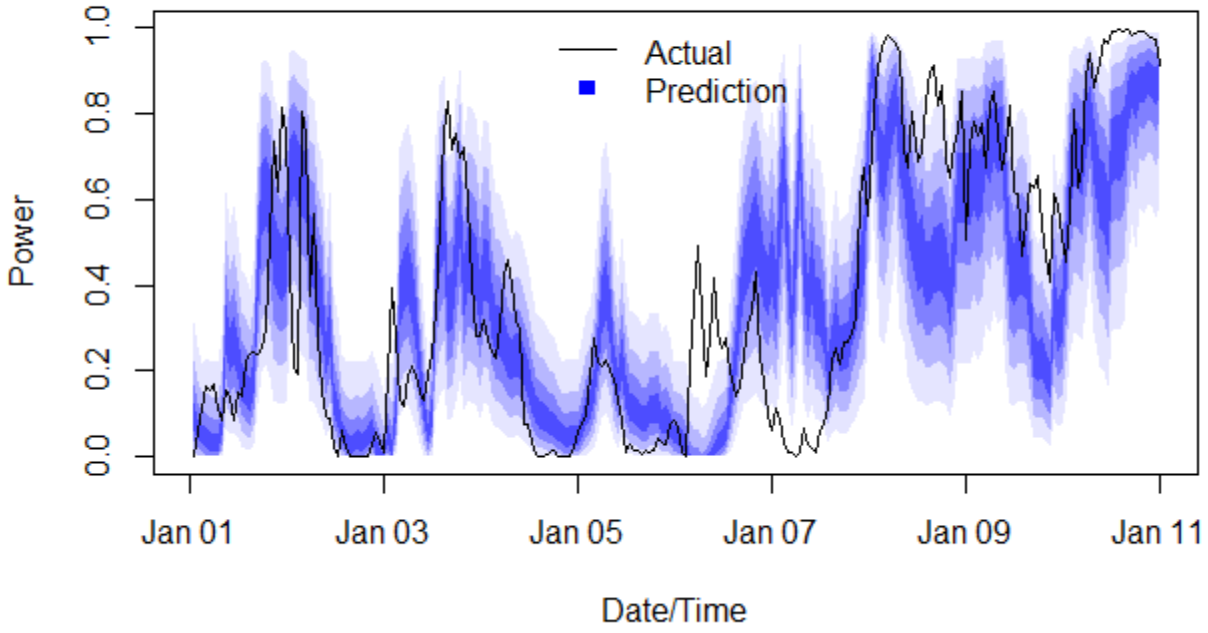


3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: GAMLSS

$$y_t \sim InflatedBeta(\boldsymbol{x}_t; a, b, p_1, p_2) = \begin{cases} p_1 & x = 0 \\ (1 - p_1 - p_2) \times Beta(\boldsymbol{x}_t; a, b) & 0 < x < 1 \\ p_2 & x = 1 \end{cases}$$

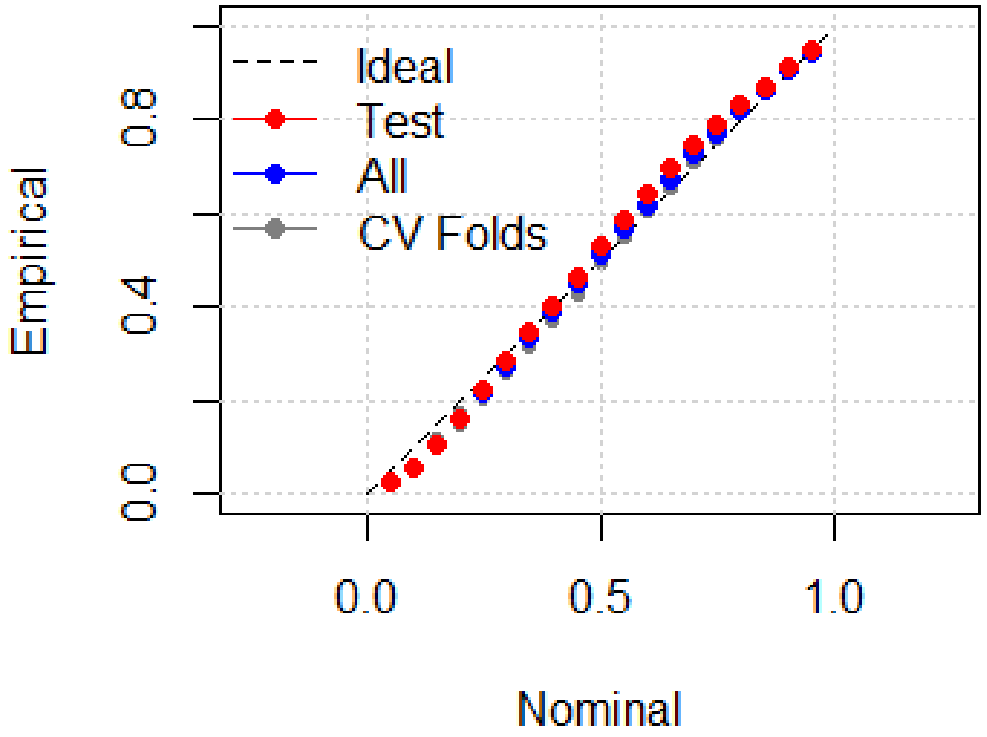


3. Linear Regression and Extensions

Generalised Additive Models for Location, Scale and Shape

Example: GAMLSS

$$y_t \sim InflatedBeta(\boldsymbol{x}_t; a, b, p_1, p_2) = \begin{cases} p_1 & x = 0 \\ (1 - p_1 - p_2) \times Beta(\boldsymbol{x}_t; a, b) & 0 < x < 1 \\ p_2 & x = 1 \end{cases}$$



3. Linear Regression and Extensions

Non-parametric Distributions

Option 3
Non-Parametric Distributions

- Complete freedom of distribution shape – great! ☺
- Infinite degrees of freedom – not so great! ☹
- Solutions:
    1. Quantile Regression (more to come)
    2. Kernel Density Estimation
        - Estimate PDF as a finite mixture of parametric functions, e.g. RBF, to approximate shape
    3. Analog Ensemble
        - Empirical distribution of historic data. May be conditional by selecting kNN historic observations to form PDF

3. Linear Regression and Extensions

Quantile Regression

Quantile Loss (Pinball) for τ^{th} quantile:

$$L_\tau(\epsilon_1, \dots, \epsilon_T) = \frac{1}{T}\sum_{\epsilon_i \geq 0} \tau\epsilon_i + \frac{1}{T}\sum_{\epsilon_i < 0} (\tau - 1)\epsilon_i$$

E.g. 10%-quantile:

$$L_{0.1}(\epsilon_1, \dots, \epsilon_T) = \frac{1}{T}\sum_{\epsilon_i \geq 0} 0.1 \times \epsilon_i + \frac{1}{T}\sum_{\epsilon_i < 0} 0.9 \times \epsilon_i$$

To minimise $L_{0.1}$ a greater number of positive ϵs relative to negative is encouraged.

3. Linear Regression and Extensions

Quantile Regression

Solution – for linear quantile regression, i.e.

$$q_{\tau,t} = \boldsymbol{x}_t^\top \boldsymbol{\beta}_\tau$$

is

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left(\frac{1}{T} \sum_{\epsilon_i \geq 0} \tau (q_{\tau,t} - \boldsymbol{x}_t^\top \boldsymbol{\beta}_\tau) + \frac{1}{T} \sum_{\epsilon_i < 0} (\tau - 1)(q_{\tau,t} - \boldsymbol{x}_t^\top \boldsymbol{\beta}_\tau) \right)$$

- Minimising quantile loss can be formulated as a linear programming problem and solved using conventional solvers
- Or iteratively using ensemble learning techniques

4. Decision Trees and Ensemble Learning

Contents

What I will cover in this section:

- Gradient Boosting
- Decision Trees
  - Gradient Boosting Trees
  - Overview of Extensions
- Bagging and Random Forest
- Analog Ensemble

4. Decision Trees and Ensemble Learning

Gradient Boosting

The Concept: Boosting

We have some simple function, $F_1(x)$, for predicting y. Can we boot it's performance by fitting another simple function, $h(x)$, to it's residuals?

$$y_t = F_2 = F_1(x_t) + \gamma h(x_t)$$

$$\underset{h,\gamma}{\operatorname{argmin}} \left[\sum_t L\left(y_t, F_1(x_t) + \gamma h(x_t)\right) \right]$$

Often finding $h(x)$ itself is not possible/practical...

4. Decision Trees and Ensemble Learning

Gradient Boosting

The Concept: *Gradient* Boosting

Take a Newton step towards lower values of loss function, i.e. gradient descent:

$$F_n(x_t) = F_{n-1}(x_t) - \gamma_n \sum_t \frac{\partial L\left(y_t, F_{n-1}(x_t)\right)}{\partial F_{n-1}}$$

Now rather than some unknown function we have new *data* called pseudo-residuals:

$$r_{n-1,t} = \left.\frac{\partial L\left(y, F\ (x)\right)}{\partial F(x)}\right|_{F(x)=F_{n-1}(x_t)}$$

Fit a new weak learner $h_n(x)$ to the pseudo-residuals $\{r_{n-1,t}, x_t\}$

4. Decision Trees and Ensemble Learning

Gradient Boosting

Gradient Boosting Algorithm

1. Calculate pseudo-residuals $r_{n-1,t} = \left.\frac{\partial L(y,F(x))}{\partial F(x)}\right|_{F(x)=F_{n-1}(x_t)}$

2. Estimate a weak learner $h_n(x)$ for the pseudo-residuals $\{r_{n-1,t}, x_t\}$

3. Calculate step size:

$$\gamma_n = \underset{\gamma}{\operatorname{argmin}} \left[\sum_t L\left(y_t, F_{n-1}(x_t) + \gamma r_{n-1,t}\right) \right]$$

4. Update model:

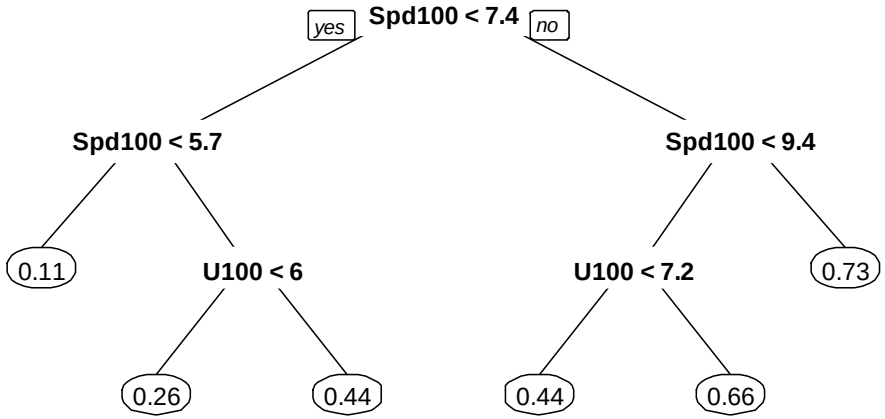
$$F_n(x) = F_{n-1}(x) + \gamma_n h_n(x)$$

4. Decision Trees and Ensemble Learning

Decision Trees

Trees: An Alternative to Linear Models

- Model dataset by partitioning using simple “rules”

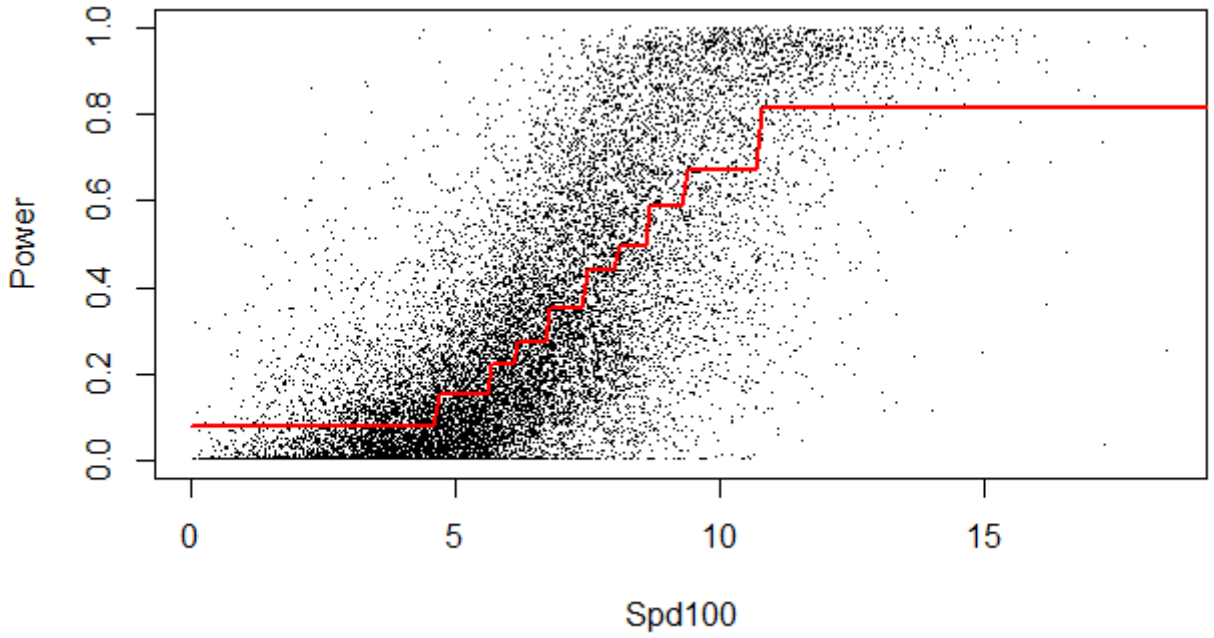


4. Decision Trees and Ensemble Learning

Decision Trees

Trees: An Alternative to Linear Models

- Model dataset by partitioning using simple “rules”

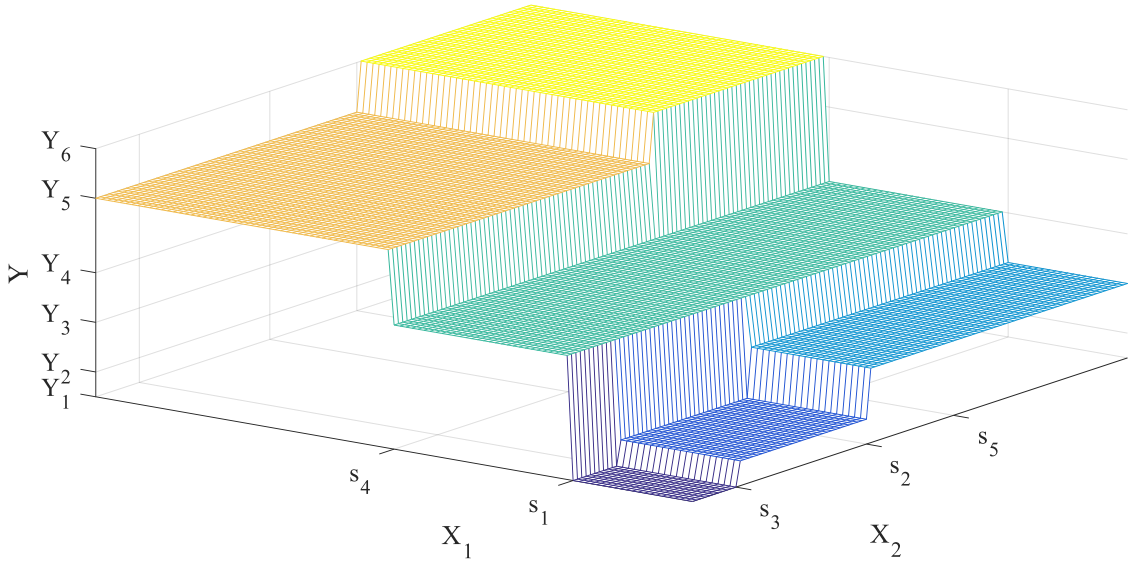


4. Decision Trees and Ensemble Learning

Decision Trees

Trees: An Alternative to Linear Models

- Properties in higher dimensions: “steps” rather than inclined planes of linear models



4. Decision Trees and Ensemble Learning

Ensemble Methods

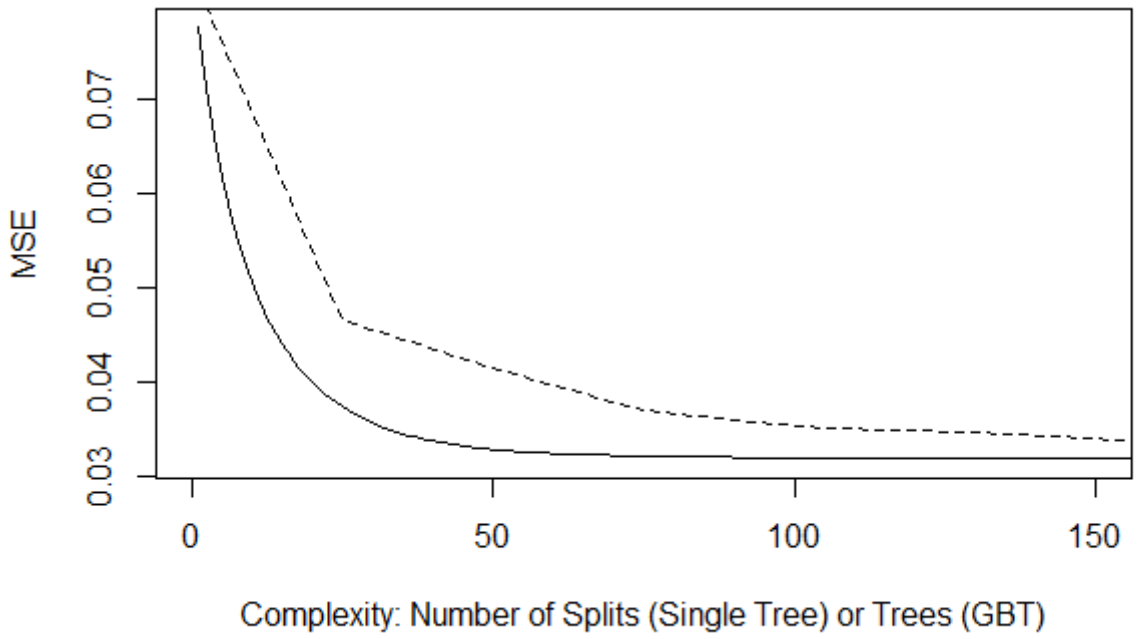
Trees: An Alternative to Linear Models

- Model dataset by partitioning using simple “rules”
- Basic process is computationally light making extensions attractive:
  - Boosting – final model = weighted sum of many *“weak learners”*
  - Bagging (“**B**ootstrap **agg**regat**ing**) – final model = weighted sum of many learners fit to subsamples of training data

4. Decision Trees and Ensemble Learning

Gradient Boosting

Boosted Simple Trees vs Single Complex Tree



Comparison of learning between single, complex decision tree and gradient boosted tree of stumps (simple decision trees). Both methods learn by sequentially increasing complexity.

4. Decision Trees and Ensemble Learning

Gradient Boosting

Gradient Boosting Algorithm: Extensions

- Stochastic Gradient Descent
  - Bagging (next section) to fit each new weak learner to random sections of training data (sampled with replacement)
  - Standard in most implementations
  - Reduce γ_n by some factor to control convergence

- Extensions/Implementations for Trees
  - XGBoost: Boosting considering a second-order Taylor expansion of the loss function. NB: loss function must be twice differentiable – can't be used for absolute error or quantile regression
  - Light GBM & CatBoost: Alternative tree growth algorithm for fast implementation on large datasets

4. Decision Trees and Ensemble Learning

Gradient Boosting

GBT for Quantile Regression

- Separate GBT required for each quantile:
  - Different hyper-parameters may be optimal for different quantiles
  - Quantile crossing may emerge, re-ordering required

- Typical hyper-parameters:
  1. Number of trees/boosting iterations
  2. Learning Rate
  3. Interaction depth (number of splits in each tree)
  4. Bagging fraction
  5. Minimum number of data points per leaf
  6. …
  7. …

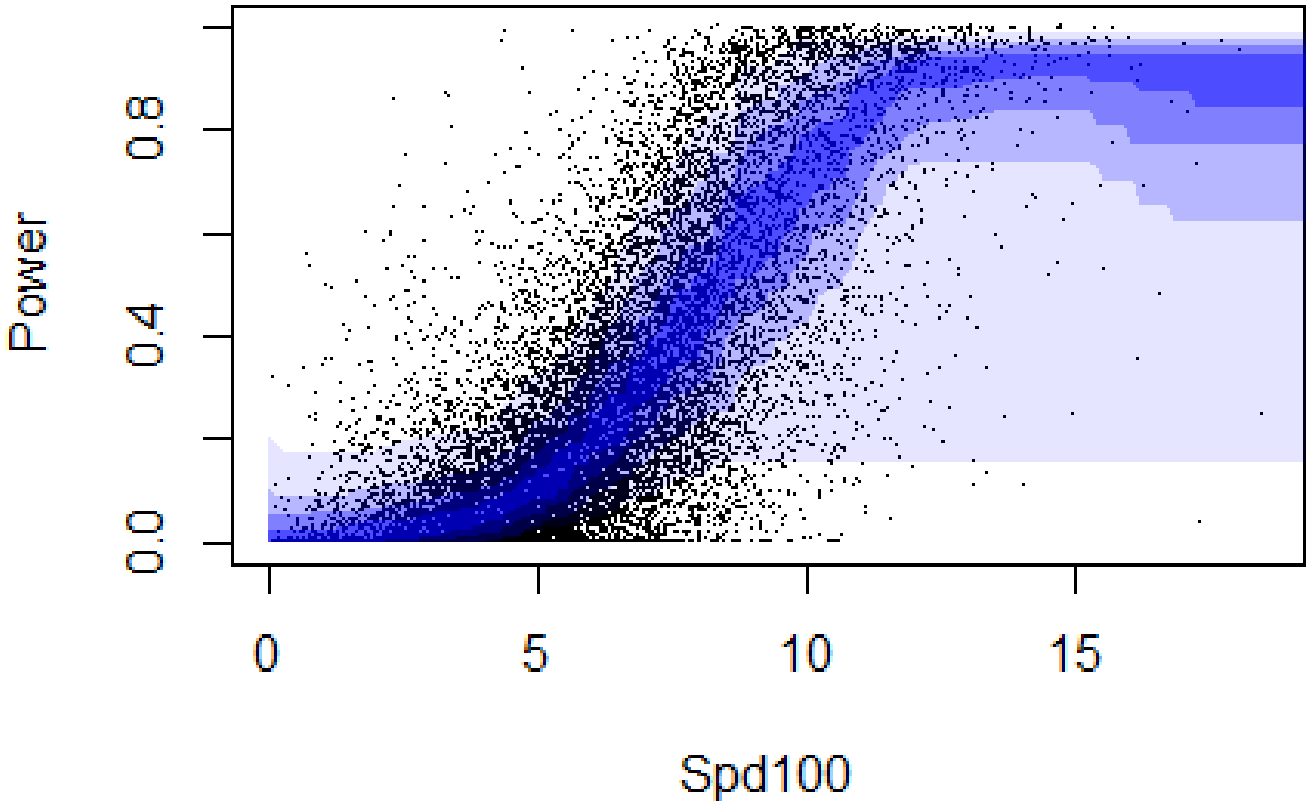
Top 3 estimated by grid search or similar

Others by judgement and trial and error.
Not so important for final results

4. Decision Trees and Ensemble Learning

Gradient Boosting

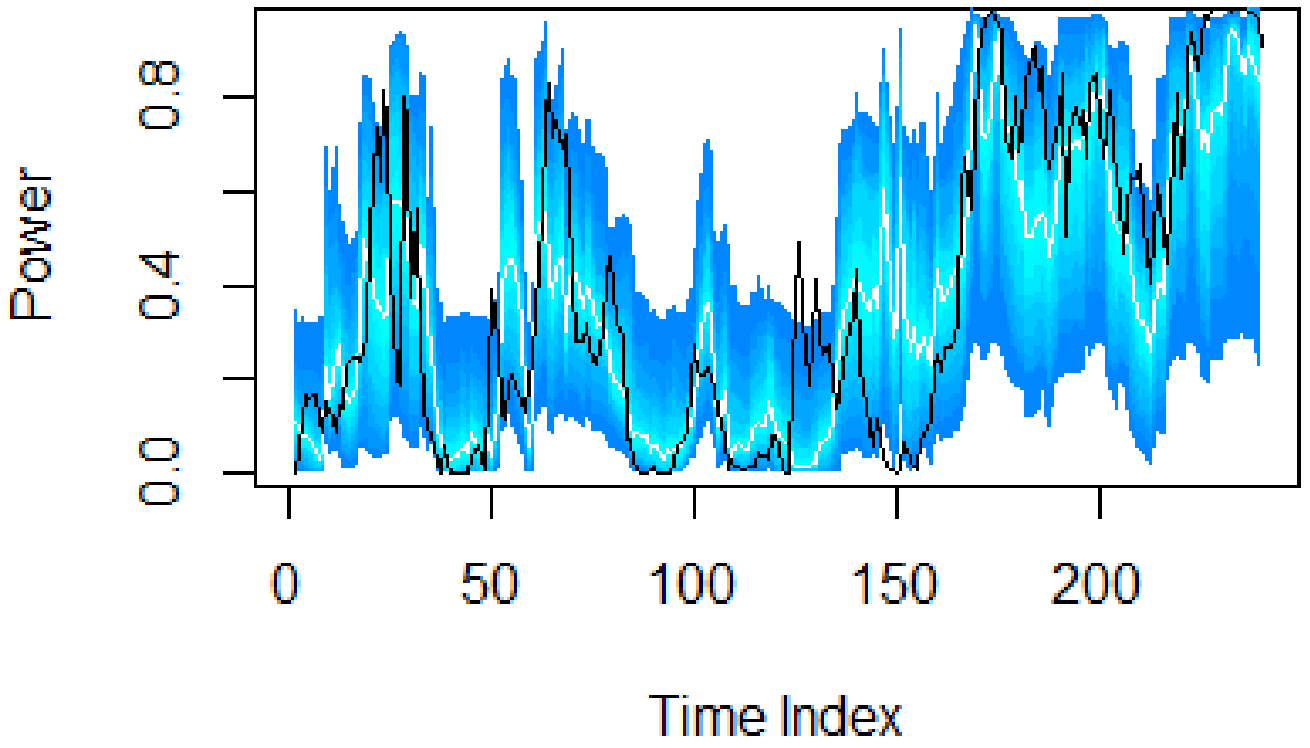
GBT for Quantile Regression



4. Decision Trees and Ensemble Learning

Gradient Boosting

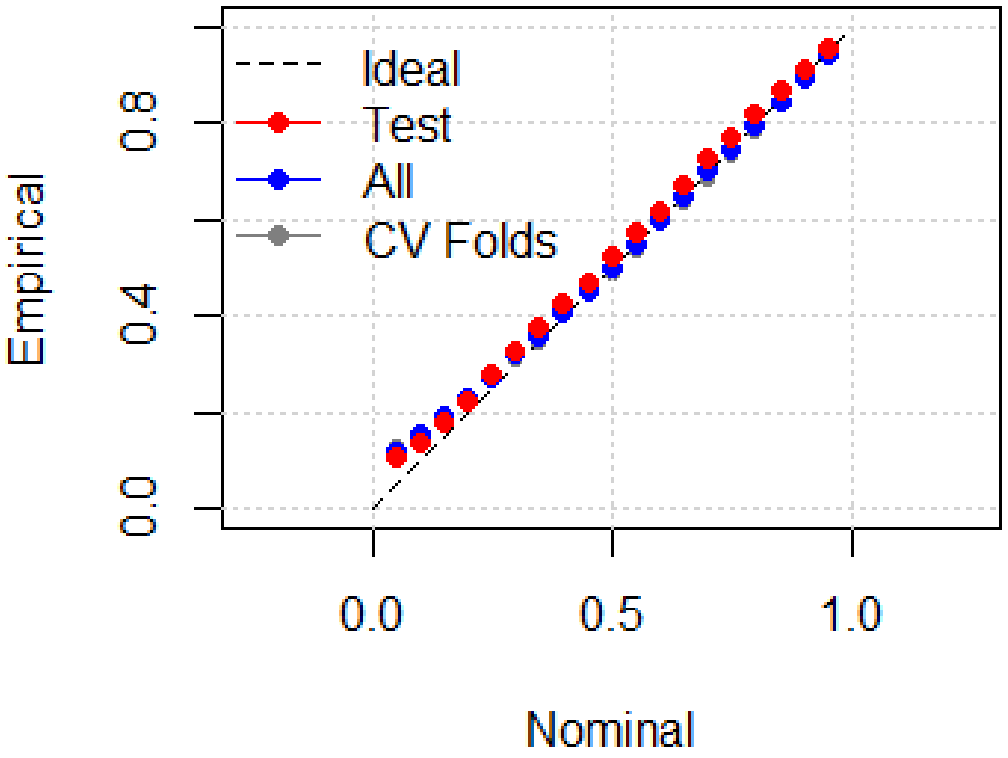
GBT for Quantile Regression



4. Decision Trees and Ensemble Learning

Gradient Boosting

GBT for Quantile Regression



4. Decision Trees and Ensemble Learning

Gradient Boosting

GBT and Feature Selection

- Gradient boosting trees perform a kind of regularisation…
  - Each simple tree only splits for the features which add the most value to the model
  - Features that contribute little are unlikely to form a split and therefore are have less impact on the model
  - Similar behaviour to LASSO…

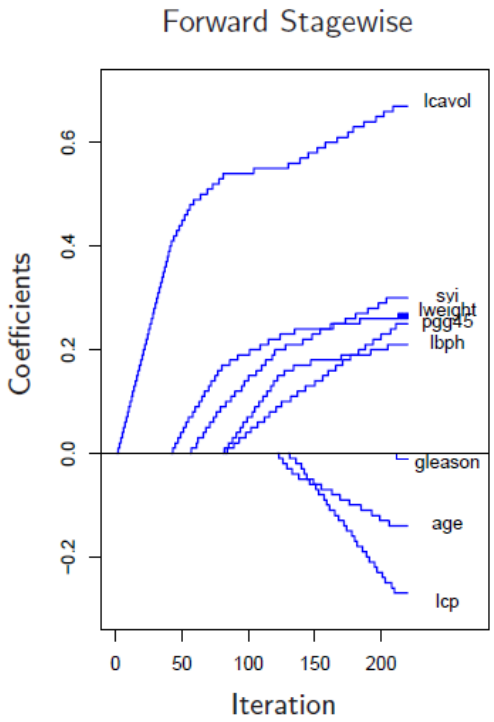
4. Decision Trees and Ensemble Learning

Gradient Boosting

GBT and Feature Selection

Forward Stagewise is a boosted linear regression algorithm:

- At each iteration, a single regression parameter is increased my a fixed amount
- The chosen parameter-feature pair is that which would have the lowest MSE if used to predict residuals in OLS



Credit: Trevor Hastie, Robert Tibshirani,Jerome Friedman,"**The Elements of Statistical Learning:** Data Mining, Inference, and Prediction", Second Edition

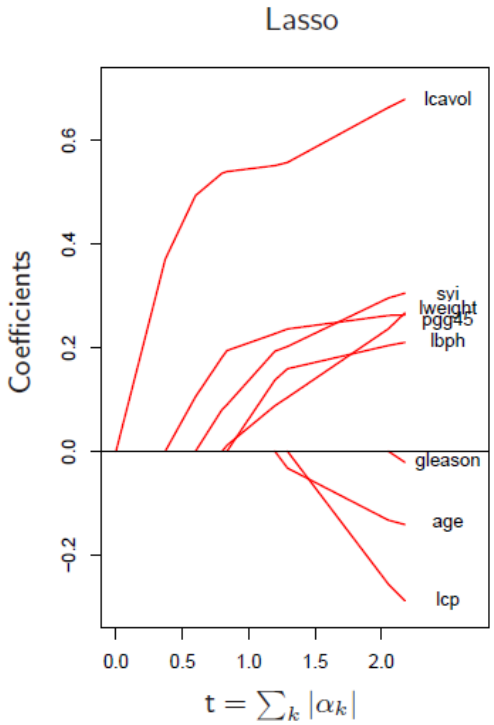
4. Decision Trees and Ensemble Learning

Gradient Boosting

GBT and Feature Selection

Compare regularisation paths to those of LASSO:

- Right: Parameter values vs sum of absolute parameter values
- Parameters estimated for a range of λ (LASSO penalty weight)



Credit: Trevor Hastie, Robert Tibshirani,Jerome Friedman,"**The Elements of Statistical Learning:** Data Mining, Inference, and Prediction", Second Edition

4. Decision Trees and Ensemble Learning

Random Forest

Random Forest: Concept

Estimate multiple decision trees and take the consensus as the final output.

Key ideas:

1. Fit each predictor to a random subset of training data
    - Not unique to random forest, also called bootstrap aggregation

2. When fitting trees, only consider a random subset of features t each candidate split, “feature bagging”.

4. Decision Trees and Ensemble Learning

Random Forest

Random Forest: Usage for Quantile Regression

- Separate forest required for each quantile:
  - Different hyper-parameters may be optimal for different quantiles (less so than for GBT)
  - Quantile crossing may emerge, re-ordering required

- Typical hyper-parameters:
  1. Number of trees in forest
  2. Sample bag fraction
  3. Feature bag fraction
  4. Minimum number of data points per leaf
  5. …
  6. …

Top 3 estimated by grid search or similar

Others by judgement and trial and error.
Not so important for final results

4. Decision Trees and Ensemble Learning

Random Forest

Random Forest: Notes

- Random Forest mitigates over-fitting by:
  - reducing model variance (by averaging result of many trees)
  - …without compromising bias.
- Preference for large forest of complex trees
  - Link to k-NN or “analog ensemble” (next slide)
- Few variants compared to boosted trees as tree growth algorithm is the defining feature of RF
- RF can be post-processed with LASSO to improve performance in some cases

4. Decision Trees and Ensemble Learning *(not really!)*

Analog Ensemble

Analog Ensemble

1. Find the *k* feature vectors in the training data that are most similar to new feature vector.
2. Use weighted sum of associated labels/outputs as prediction. Use empirical distribution of labels/outputs as uncertainty forecast.

- Simple and powerful method!
- No model to train, though choice of *k* and distance measure is important
- Limited by number of features – performance deteriorates with low quality or large number of features
- Relatively high computation cost for operational forecasting

5. Practical Example

Which approach should I choose?

Things I haven't talked about today but you might be interested in:

- Feature Engineering
  - *Next Lecture!*
- Neural Networks
  - *Tomorrow's lectures!*
- Conditional Kernel Density Estimation
  - Estimate conditional density as a linear combination of kernel functions
- Markov chains
- Support Vector Machines
- Mixture models, hidden Markov models and the EM algorithm
  - For unobserved/unlabelled regimes – really powerful stuff!

5. Practical Example

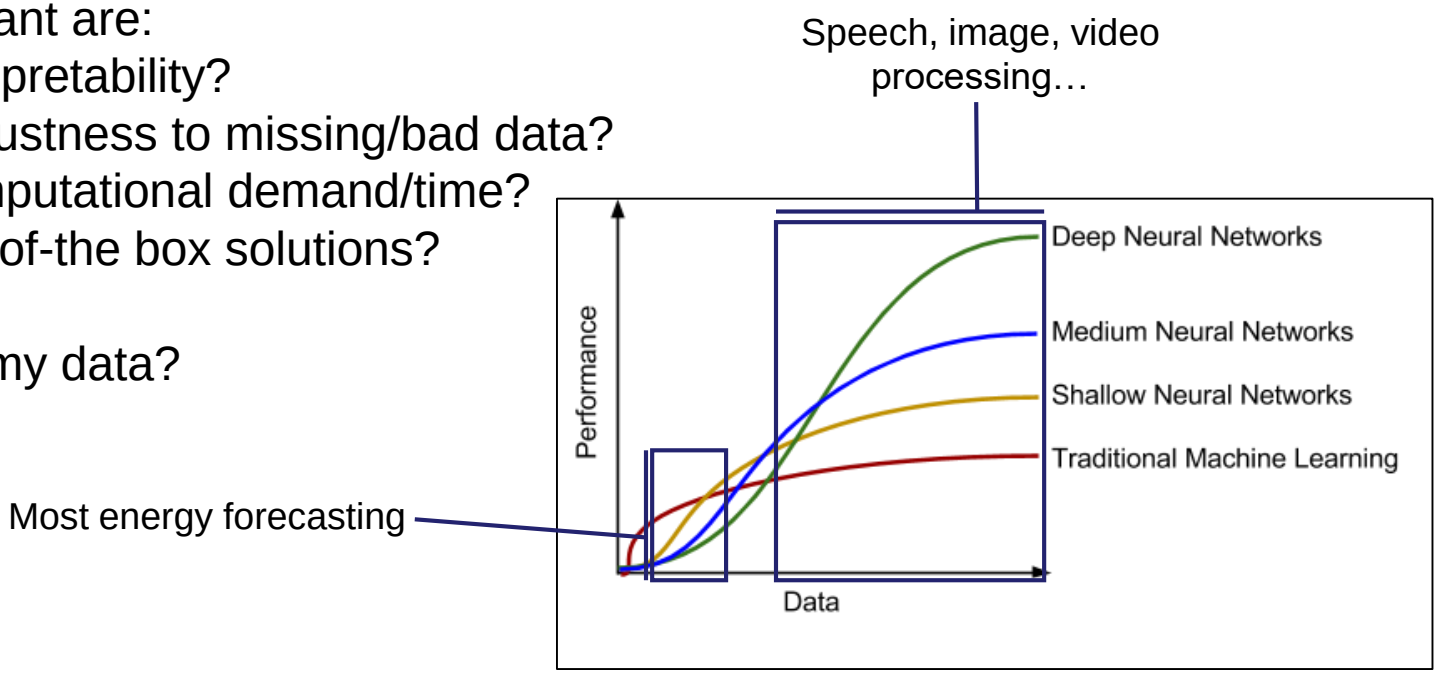
Which approach should I choose?

Some questions to ask yourself:

- Which approach do I understand best? (Do I have time to learn a new method?)
    - *Is performance going to be dominated by feature engineering anyway? (Next lecture!)*

- How important are:
    - Interpretability?
    - Robustness to missing/bad data?
    - Computational demand/time?
    - Out-of-the box solutions?

- How big is my data?



5. Practical Example

Tuning a GBT for Wind Power Forecasting

Practical Guide to Tuning a GBT

1. Data Preparation
2. Training and Validation Set-up
3. Initial Tuning
4. Fine Tuning
5. Final Evaluation

5. Practical Example

Tuning a GBT for Wind Power Forecasting

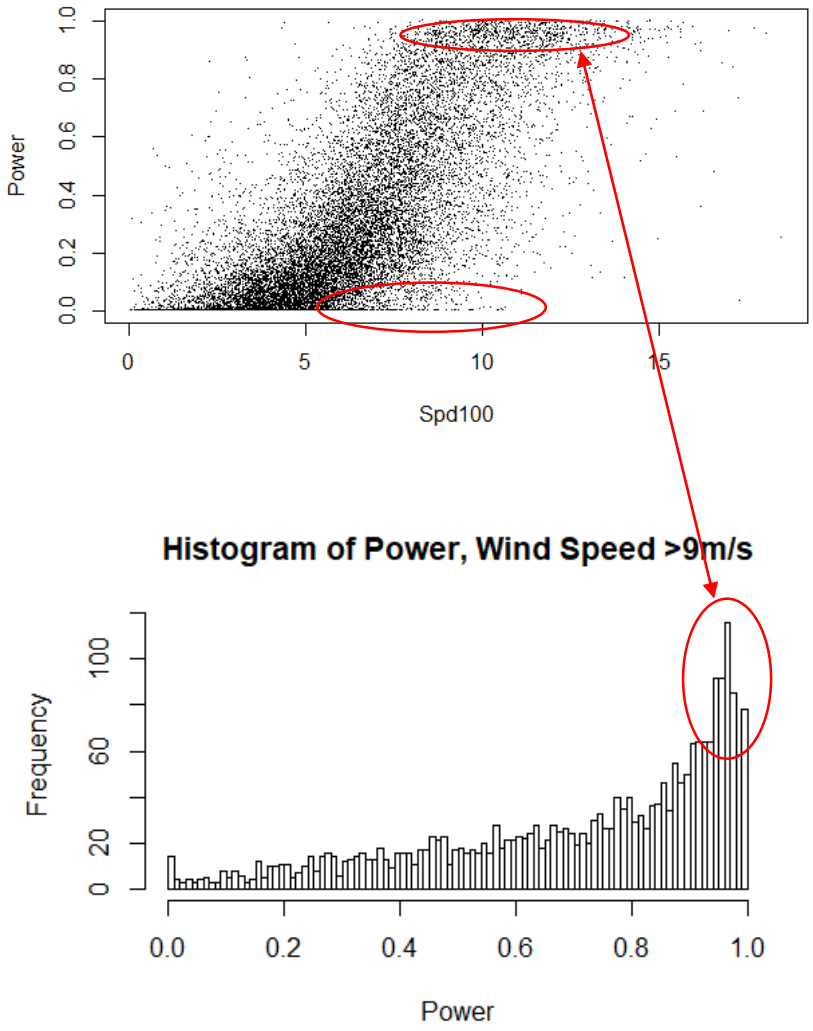
1. Data Preparation

Remove/adjust data which are not representative of the process you are modelling:

- Curtailment
- Reduced capacity
- Meter errors

Other considerations:

- Forecast horizon – performance on NWP changes with horizon…
    1. Learn different model for different horizons
    2. Create explanatory features
- Slow changes in training data:
    1. Performance degradation/restoration
    2. Change in local geography, e.g. forestry
    - ➢ Create explanatory features



5. Practical Example

Tuning a GBT for Wind Power Forecasting

2. Training and Validation Set-up

Evaluation metrics: Sharpness subject to reliability?

- Quantile loss of individual quantiles and overall

Cross-validation

- Divide training set into *k*-fold
- Choose *k* carefully: large is good but at computational expense!
- Some applications will favour random samples for CV – beware training on data which are highly (unrealistically) correlated to test data!

Train	Train	Train	Validation
Train	Train	Validation	Train
Train	Validation	Train	Train
Validation	Train	Train	Train

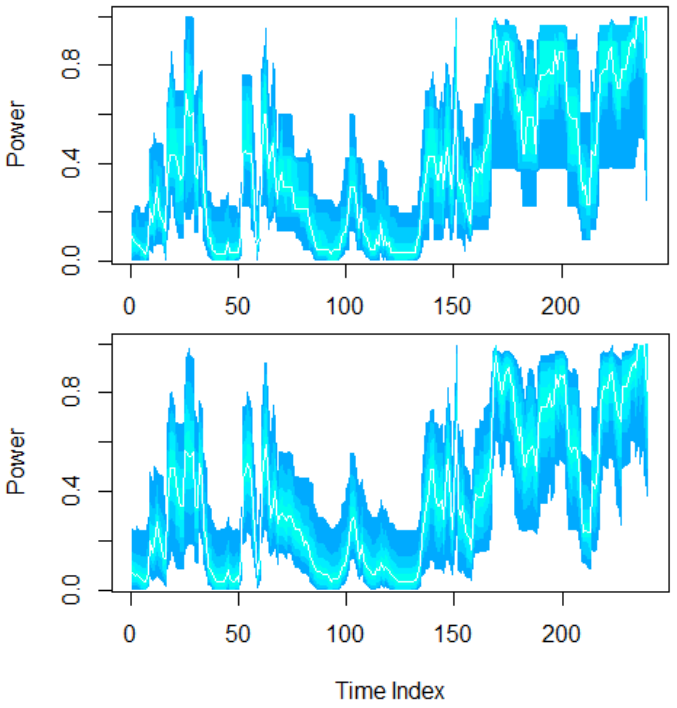
5. Practical Example

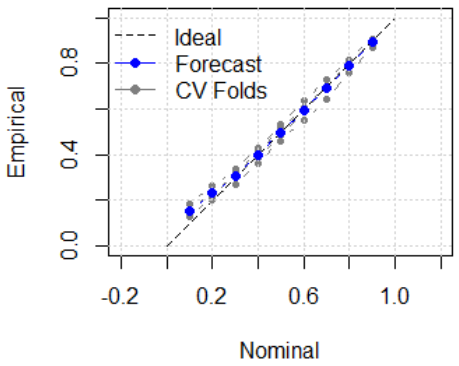
Tuning a GBT for Wind Power Forecasting

3. Initial Tuning

Get a rough idea of what is going to work:

- Sensible starting values and ranges for hyper-parameters
- Feature Engineering! (*Next Lecture)*
- Sensitivity of different quantiles to different parameters
  - Should I tune them separately?
  - Which parameters should I fine tune?
- Plot results!
  - Check behaviour is sensible!
  - Can you identify any failings? E.g. poor performance at low/high wind speed etc…



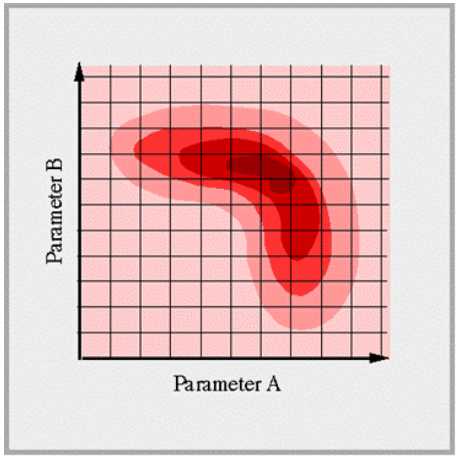


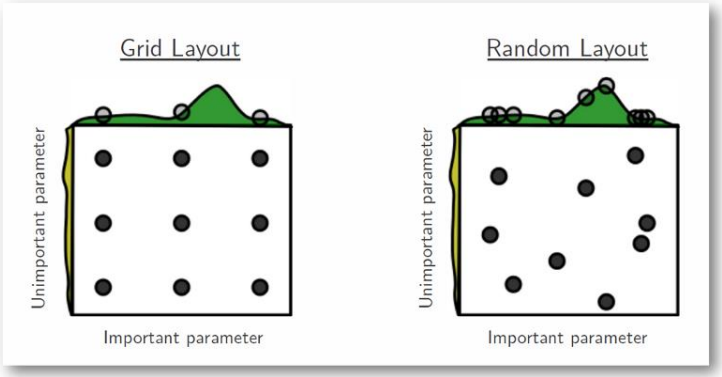
5. Practical Example

Tuning a GBT for Wind Power Forecasting

4. Fine Tuning

- Set-up an automated grid search (or better) to find best combination of hyper-parameters
    - Alternatives to grid search can be much faster and more effective
    - Parallelisable
- Be smart but patient
- Re-order crossed quantiles if required





James Bergstra, Yoshua Bengio, Random Search for Hyper-Parameter Optimization,Journal of Machine Learning Research, 13, 2012

5. Practical Example

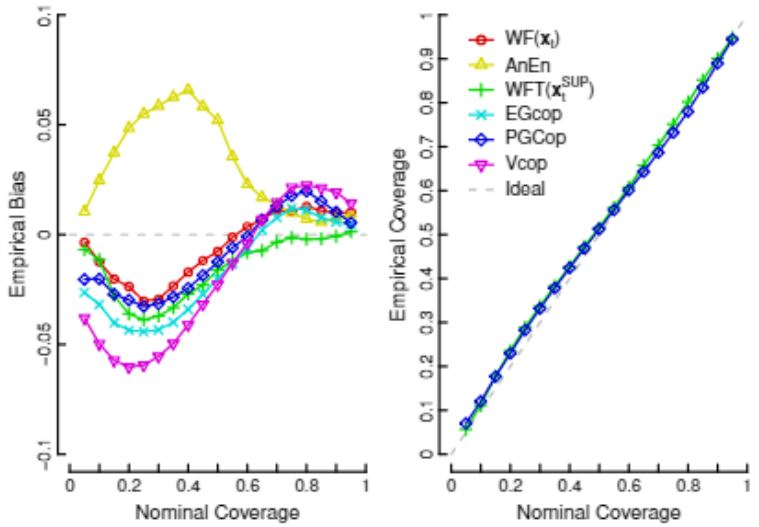
Tuning a GBT for Wind Power Forecasting

5. Evaluation

- Produce final result for previously unseen Test Data

- Analyse performance, overall and under different conditions:
  - Reliability
  - Quantile Loss
  - Sharpness
  - Decision-based metrics?

- Compare to other methods:
  - Simple benchmarks
  - Competitive benchmarks



References

Text Books on Statistical Learning

1. Trevor Hastie, Robert Tibshirani,Jerome Friedman,"The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Second Edition
2. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, "An Introduction to Statistical Learning"
3. Simon N. Wood, "Generalized Additive Models: An Introduction with R," Second Edition

Weather and Energy

1. Weather & Climate Services for the Energy Industry, Editors: Troccoli, Alberto (open access e-book), Palgrave Macmillan, 2018
2. Weather Matters for Energy, Editors: Troccoli, A., Dubus, L., Haupt, S.E., Springer, 2014

Use of Uncertainty Forecasts in Power System Operation

1. Bessa, R.J.; Möhrlen, C.; Fundel, V.; Siefert, M.; Browell, J.; Haglund El Gaidi, S.; Hodge, B.-M.; Cali, U.; Kariniotakis, G. Towards Improved Understanding of the Applicability of Uncertainty Forecasts in the Electric Power Industry. Energies 2017, 10, 1402. doi: 10.3390/en10091402
2. J. Dobschinski, R. Bessak, P. Du, K. Geisler, S.E. Haupt, M. Lange, C. Möhrlen, D. Nakafuji and M. de la Torre Rodriguez, Uncertainty Forecasting in a Nutshell: Prediction Models Designed to Prevent Significant Errors, IEEE Power and Energy Magazine, vol. 15, no. 6, pp. 40-49, Nov.-Dec. 2017, doi: 10.1109/MPE.2017.2729100

For specific methods see references in the two papers above.