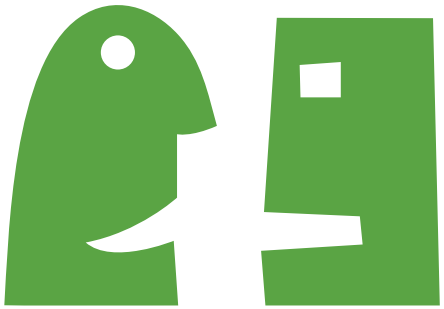
Wagging speech by the tail

The case for robust data generation



Natural Speech Technology

Edinburgh – Cambridge – Sheffield

Gustav Eje Henter

Synopsis

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions

Synopsis

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions
2. This is due to a largely unrecognised mismatch between parameter estimation and output generation

Synopsis

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions
2. This is due to a largely unrecognised mismatch between parameter estimation and output generation
   - Maximum likelihood fits the outskirts (tails) of the data distribution, where the bad datapoints sit

Synopsis

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions
2. This is due to a largely unrecognised mismatch between parameter estimation and output generation
   - Maximum likelihood fits the outskirts (tails) of the data distribution, where the bad datapoints sit
   - The distribution peak – "typical data" – is assigned the lowest importance, even though that is what is used to generate output

Synopsis

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions
2. This is due to a largely unrecognised mismatch between parameter estimation and output generation
   - Maximum likelihood fits the outskirts (tails) of the data distribution, where the bad datapoints sit
   - The distribution peak – "typical data" – is assigned the lowest importance, even though that is what is used to generate output
3. Robust statistics can de-emphasise the tails and better describe the parts of the distribution used for generation

Synopsis

Natural Speech Technology

1. Data-generation tasks like statistical speech synthesis are sensitive to bad data and bad assumptions
2. This is due to a largely unrecognised mismatch between parameter estimation and output generation
   - Maximum likelihood fits the outskirts (tails) of the data distribution, where the bad datapoints sit
   - The distribution peak – "typical data" – is assigned the lowest importance, even though that is what is used to generate output
3. Robust statistics can de-emphasise the tails and better describe the parts of the distribution used for generation
   - This yields improved speech-sound durations in an application

Take-home message

Natural Speech Technology

Theorist:

- Generation tasks (label to observation) have different priorities from classification tasks (observation to label)
- Approaches tailored for data-generation problems is an under-explored topic in machine learning

Take-home message

Natural Speech Technology

Theorist:

- Generation tasks (label to observation) have different priorities from classification tasks (observation to label)
- Approaches tailored for data-generation problems is an under-explored topic in machine learning

Practitioner:

- Bad data and assumptions are more dangerous than you think
- Common generation setups suffer from a hidden mismatch
  - The methods work especially poorly on big/found/imperfect data
- Robust statistics can mitigate the resulting problems
- *Make sure your approach has the same priorities as you have!*

Overview

Background section

Natural Speech Technology

Setting the stage:

1. Introduction
2. Sensitivity issues
3. Error sources
4. Proposed solutions

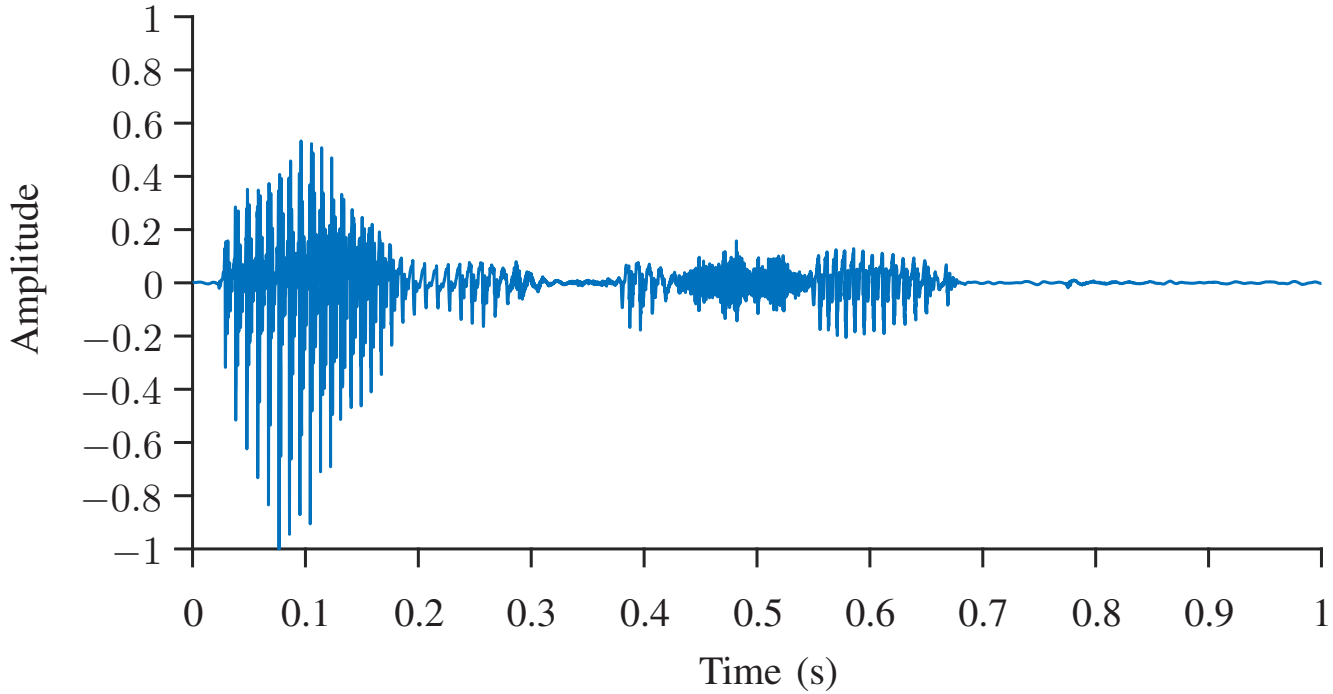
Speech synthesis

Natural Speech Technology

- Text input $\rightarrow$ speech audio output
  - Text to speech (TTS)
  - Low bitrate to high bitrate
- ...using parallel speech+text data and statistical models
  - Statistical parametric speech synthesis (SPSS)
- This will be our running example of a data-generation task

Input and output features

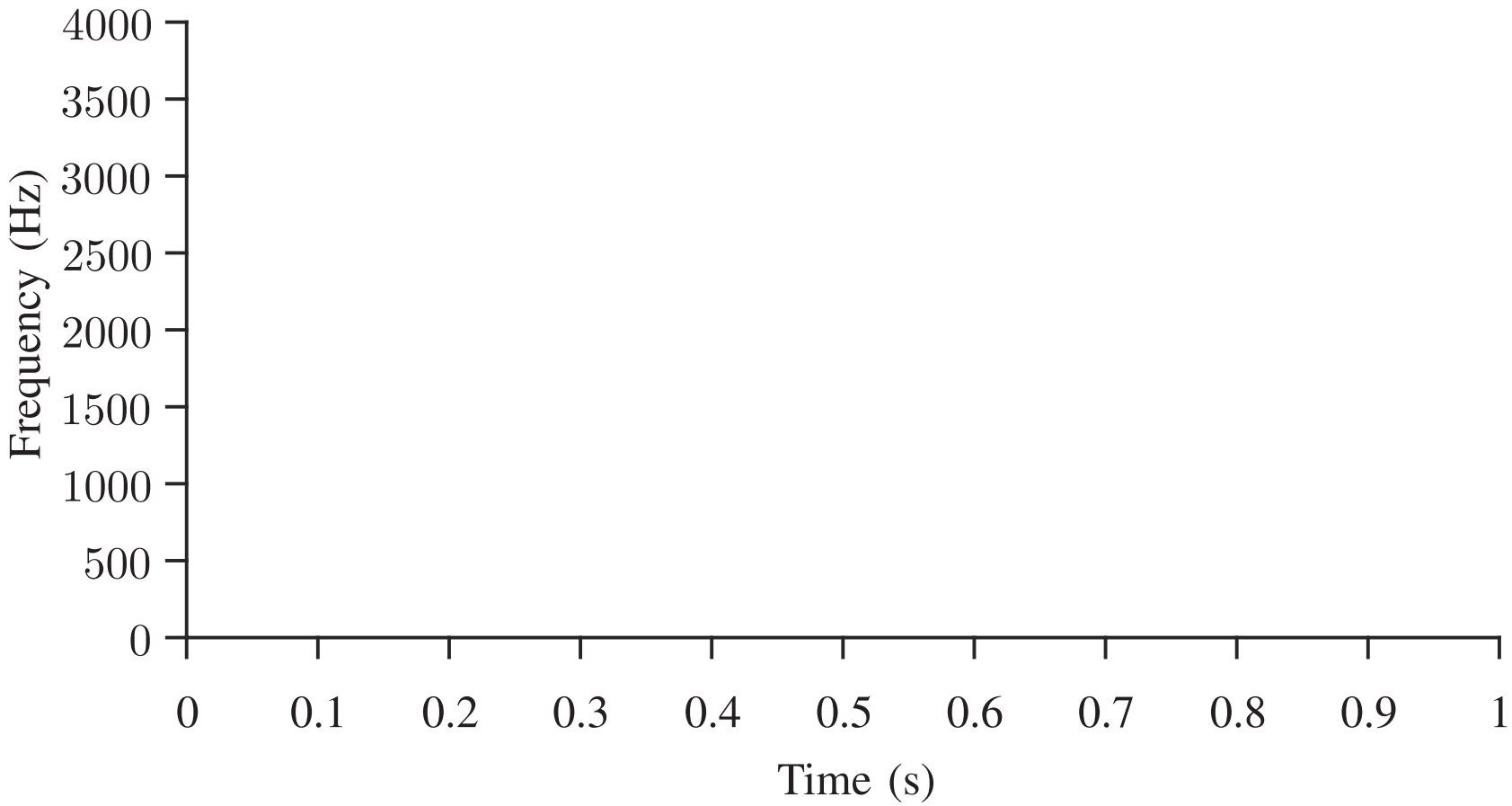
- In: Spoken text, e.g., "I was the sheep."
  - Phone sequence: "- AY W AA Z DH AH SH IY P -"
- Out: Audio waveform (16k samples per second or more), e.g.,



  - Vocoder analysis: Convert to acoustic feature vectors at 200 fps

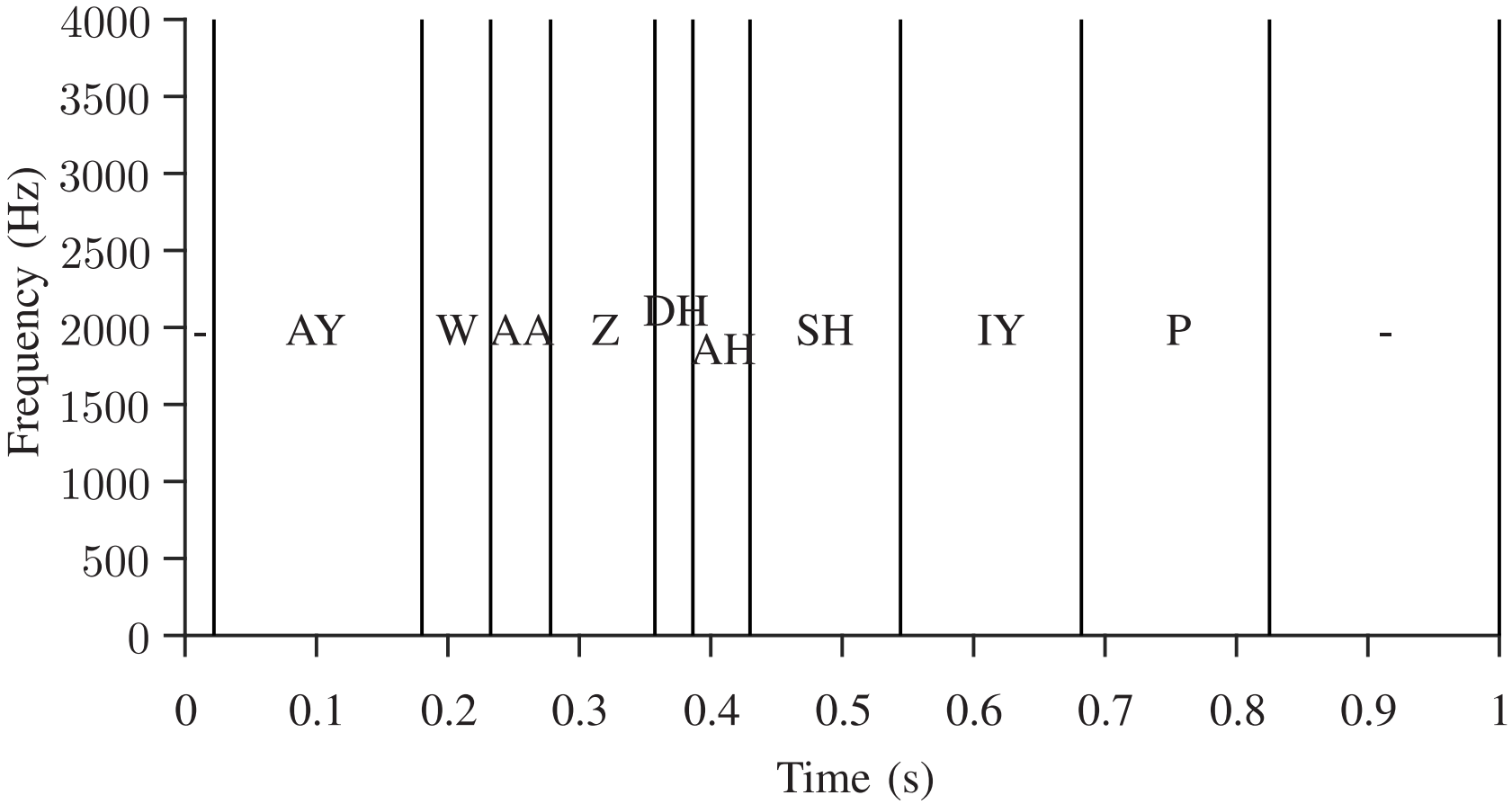
Two modelling stages

Task: Map "- AY W AA Z DH AH SH IY P -" to acoustic features



Two modelling stages

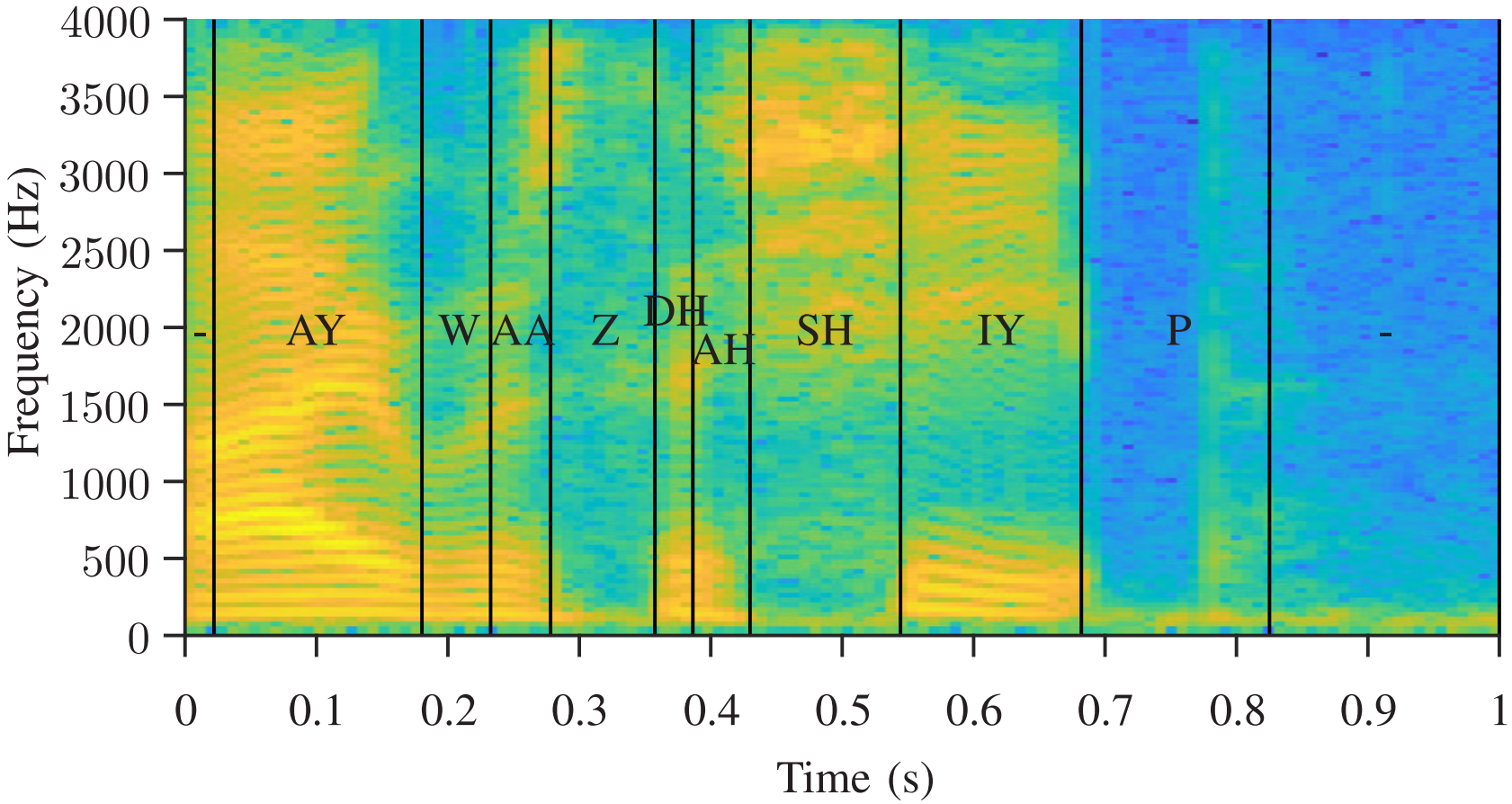
Duration model: Predicts the duration of each phone



Two modelling stages

Natural Speech Technology

Acoustic model: Fills in the acoustic features of each phone



Statistical modelling

Natural Speech Technology

Two modelling stages:

1. Duration model (phoneme-level)
2. Acoustic model (frame-level)

Common elements:

- Sequence-valued data
- Linguistic (discrete, text-based) input feature vectors $\boldsymbol{l}$
- Output feature vectors $\boldsymbol{x}$ (durations or acoustics; continuous)
- Probabilistic regression model $f_{\boldsymbol{X} \mid \boldsymbol{L}}(\boldsymbol{x} \mid \boldsymbol{l}; \boldsymbol{\theta})$
  - Unknown model parameters $\boldsymbol{\theta}$

Main steps

There are three main steps of probabilistic text-to-speech:

1. Model specification: Propose a probabilistic model
2. Training: Estimate model parameters on training data
3. Synthesis: Generate output sequences from fitted model

Training and synthesis

Natural Speech Technology

1. Training: Maximum likelihood parameter estimation (MLE)
   - Aligned training data $\mathcal{D} = \{(\boldsymbol{l}_t, \boldsymbol{x}_t)\}$

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \operatorname*{argmax}_{\boldsymbol{\theta}} \prod_{(\boldsymbol{l}_t, \boldsymbol{x}_t) \in \mathcal{D}} f_{\boldsymbol{X} \mid \boldsymbol{L}}(\boldsymbol{x}_t \mid \boldsymbol{l}_t; \boldsymbol{\theta})$$

2. Synthesis: Maximum likelihood parameter generation (MLPG)
   - Linguistic features $\{\boldsymbol{l}_t\}$ from input text

$$\{\widehat{\boldsymbol{x}}_t\} = \operatorname*{argmax}_{\{\boldsymbol{x}_t\}_t} \prod_t f_{\boldsymbol{X} \mid \boldsymbol{L}}\left(\boldsymbol{x}_t \,\middle|\, \boldsymbol{l}_t; \widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\right)$$

Training and synthesis functions appear to be well matched

Training and synthesis

Natural Speech Technology

1. Training: Maximum likelihood parameter estimation (MLE)
   - Aligned training data $\mathcal{D} = \{(\boldsymbol{l}_t, \boldsymbol{x}_t)\}$

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ML}} = \operatorname*{argmax}_{\boldsymbol{\theta}} \prod_{(\boldsymbol{l}_t, \boldsymbol{x}_t) \in \mathcal{D}} f_{\boldsymbol{X} \mid \boldsymbol{L}}(\boldsymbol{x}_t \mid \boldsymbol{l}_t; \boldsymbol{\theta})$$

2. Synthesis: Maximum likelihood parameter generation (MLPG)
   - Linguistic features $\{\boldsymbol{l}_t\}$ from input text

$$\{\widehat{\boldsymbol{x}}_t\} = \operatorname*{argmax}_{\{\boldsymbol{x}_t\}_t} \prod_t f_{\boldsymbol{X} \mid \boldsymbol{L}}\left(\boldsymbol{x}_t \,\middle|\, \boldsymbol{l}_t; \widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\right)$$

Training and synthesis functions appear (but aren't) well matched

A general output principle

Natural Speech Technology

Returning the a-posteriori most probable outcome is exceedingly common across output domains:

- Classification (label domain)
  - Bayes classifier (minimum misclassification rate)
  - Applied classifiers
- Generation and prediction (observation domain)
  - Speech (text-to-speech, voice conversion, ...)
  - Text (machine translation, captioning, ...)
  - Any predictor based on minimising mean squared error (MSE)
    - The MSE minimiser can be derived as the maximum-likelihood estimate of the mode in a fixed-variance Gaussian model

Standard choices

Natural Speech Technology

Traditional setup (Zen et al., 2007):

1. Assume output distribution $f_{\boldsymbol{X} \mid \boldsymbol{L}}$ is diagonal-covariance Gaussian $f_{\mathcal{N}}\left(\boldsymbol{x}; \boldsymbol{\mu}\left(\boldsymbol{l}\right), \operatorname{diag}\left(\boldsymbol{\sigma}^2\left(\boldsymbol{l}\right)\right)\right)$
2. Estimate means and variances using MLE
3. Use an efficient algorithm (Tokuda et al., 2000) to generate the most probable output sequence from the Gaussian
   - For Gaussian models, the mode equals the mean: $\widehat{\boldsymbol{x}} = \widehat{\boldsymbol{\mu}}$

Background section

Natural Speech Technology

Setting the stage:

1. Introduction
2. Sensitivity issues
3. Error sources
4. Proposed solutions

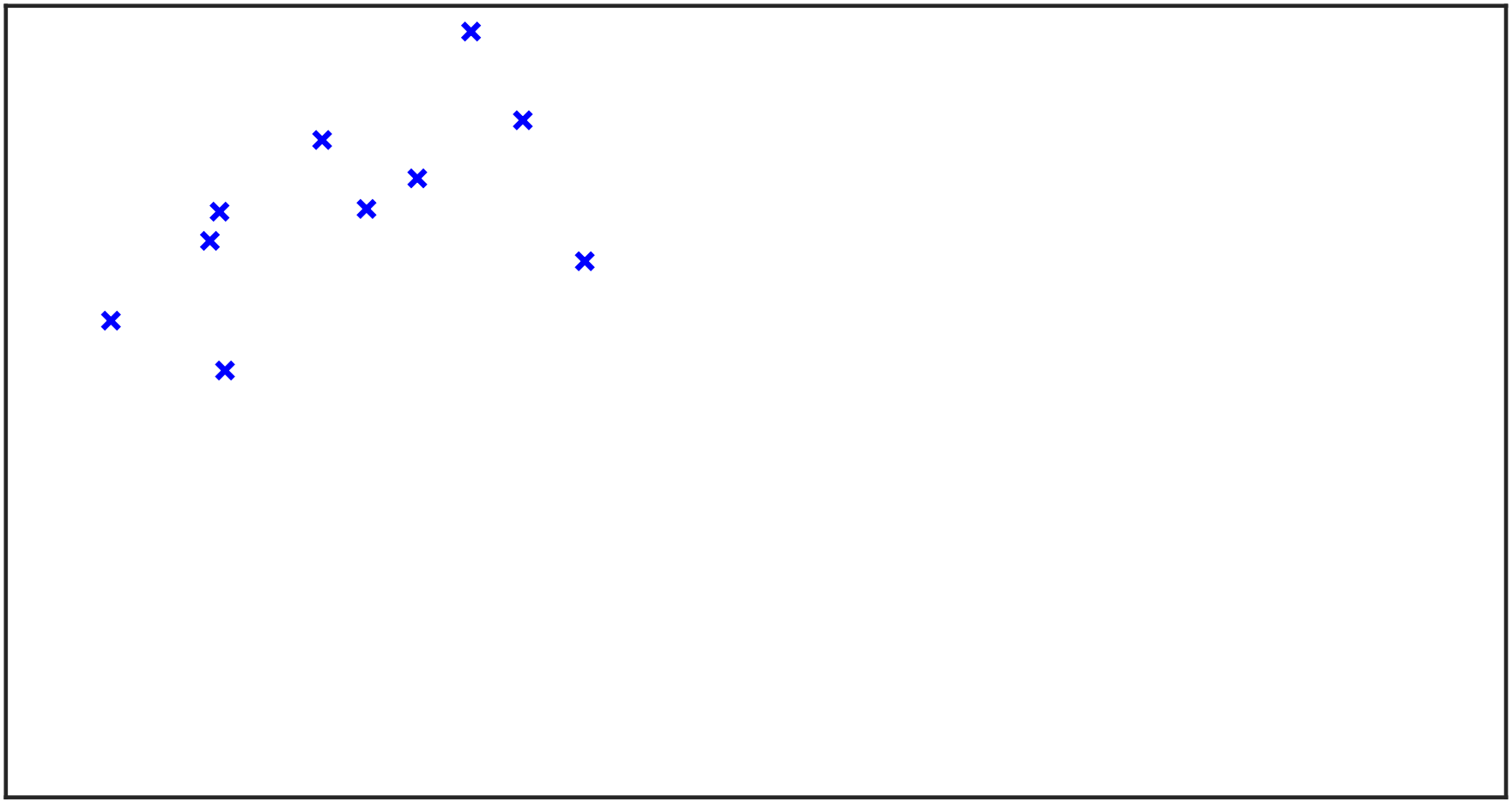
Issues

Since the generated output sounds unnatural, something is wrong in the standard approach

- Many sources contribute to unnaturalness (Henter et al., 2014; Uría et al., 2015)
- Quality degrades further if the data isn't completely pure (Yamagishi et al., 2008)
  - This necessitates careful quality control of synthesis data
  - Large, found speech databases sit unused

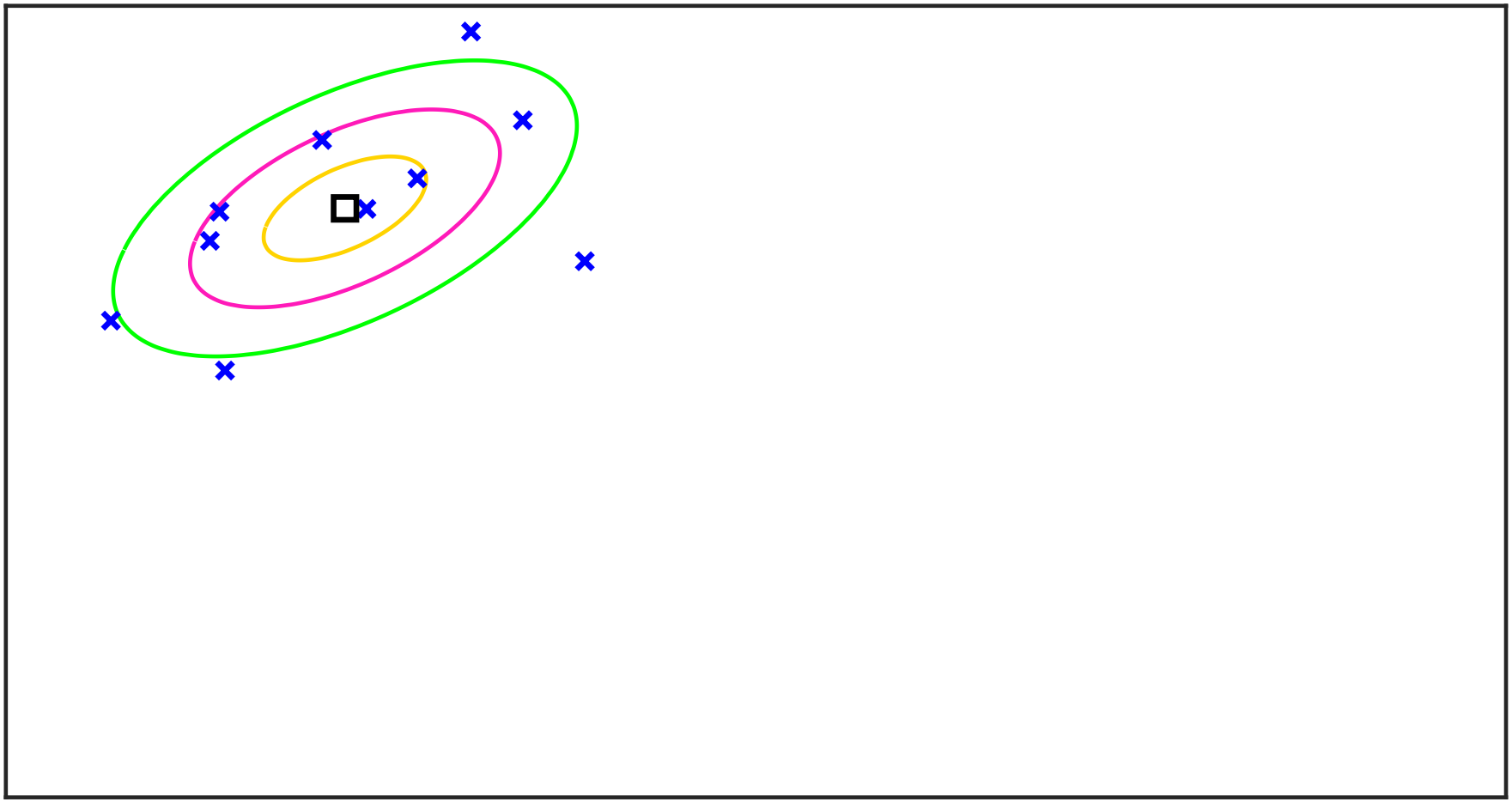
Toy example

Generate some datapoints $\mathcal{D}$



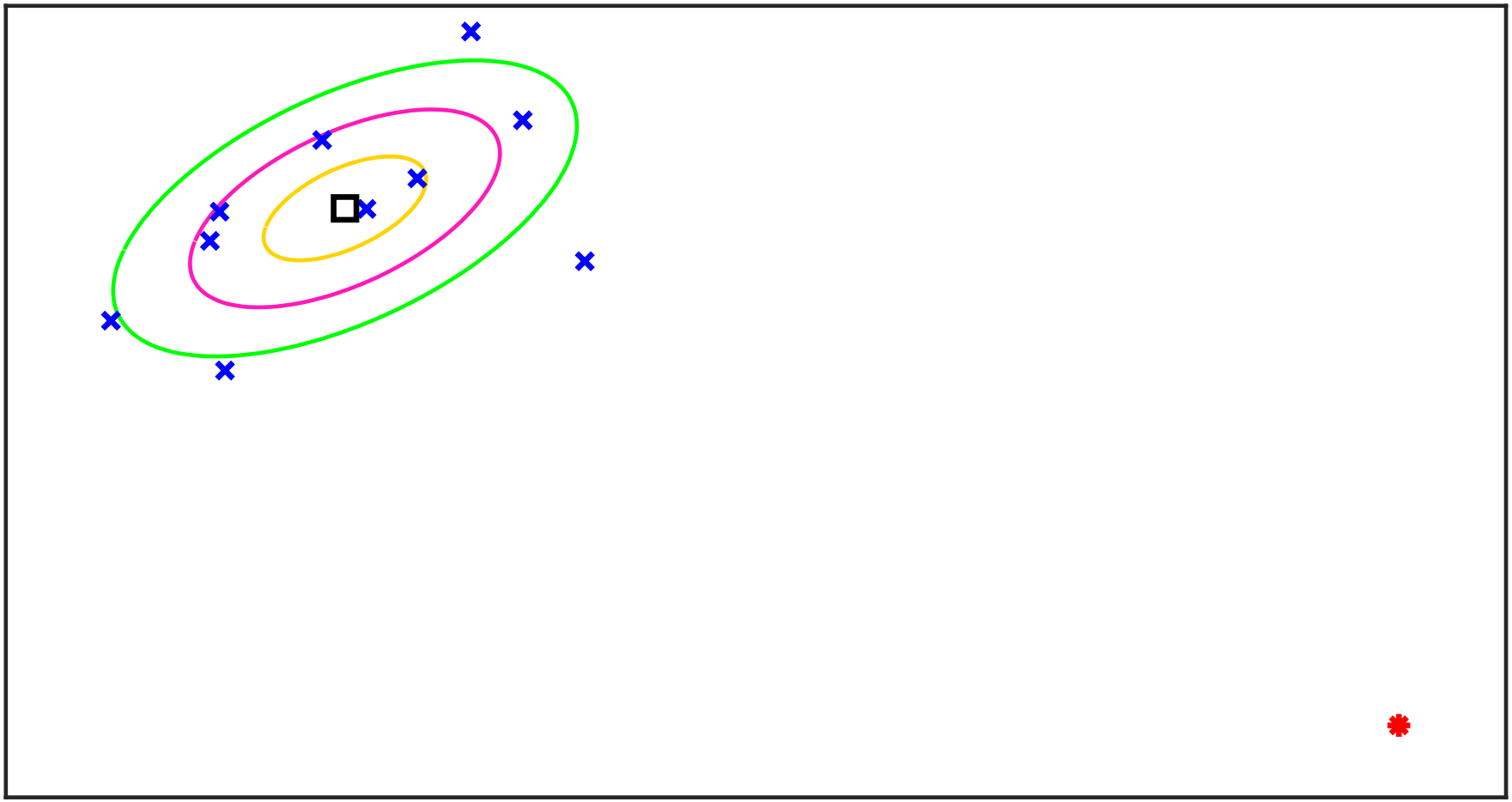
Toy example

Fit a Gaussian using maximum likelihood



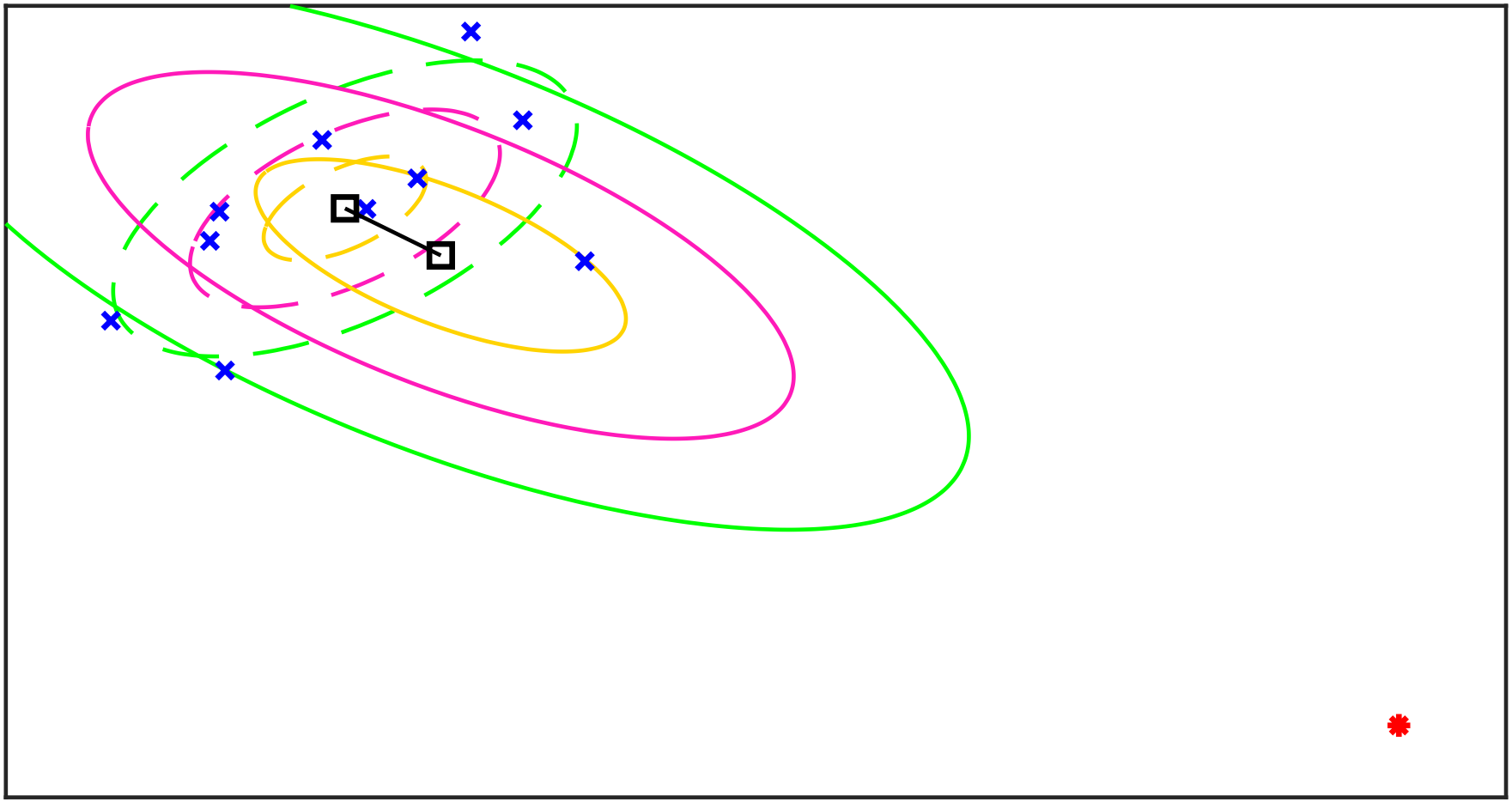
Toy example

Add an unexpected datapoint



Toy example

The maximum likelihood fit changes a lot!



What went wrong?

Natural Speech Technology

The data didn't match the model!

- The theoretical results that make MLE appealing depend crucially on a good match between model and data

Why did it have these consequences?

- If the model cannot fit the data well everywhere, we must choose which part to prioritise
- MLE prioritises the tails over the peak

Why is this a problem?

Natural Speech Technology

- All models are wrong and cannot be accurate everywhere
- MLE fits the tails of the distribution to the data
- Synthesis, meanwhile, only uses the peak of the model

$\Rightarrow$ The training is a poor match to the application!

  - "Wagging speech by the tail"

Practical consequences

Natural Speech Technology

Generally: A focus on the tails of the data $\Rightarrow$ the fitted mode is sensitive to ill-fitting points

Specifically: The mean, as used in synthesis, may not fall in a high-probability region $\Rightarrow$ output need not be like speech at all

We will see that robust methods can mitigate these issues

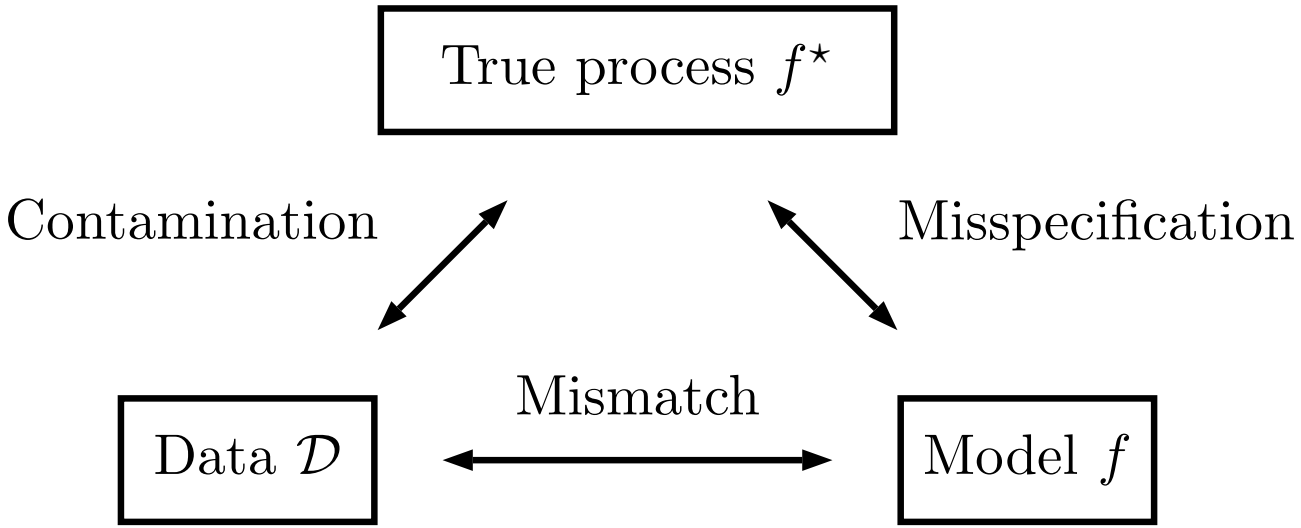
Background section

Natural Speech Technology

Setting the stage:

1. Introduction
2. Sensitivity issues
3. Error sources
4. Proposed solutions

A taxonomy of errors



(Note that model-data mismatch is different from estimation-generation mismatch)

Sources of model-data mismatch

Natural Speech Technology

- Misspecification: The true process $f^\star$ does not match the proposed model f (always a problem)
  - E.g., skewed, non-Gaussian feature distributions
- Contamination: The data $\mathcal{D}$ is not from the true process of interest $f^\star$ (big issue in large, found data)
  - Audio issues, e.g., packet loss, background noise, clipping, pronunciation errors
  - Text issues, e.g., transcription mistakes, wrong file number
  - Internal issues, e.g., out-of-vocabulary words, homograph resolution, alignment failures

Robust statistics

"Robust" can mean many things

- Here: Statistical techniques with low sensitivity to model-data mismatch (Huber, 2011)
- Think: Modelling techniques that are able to *disregard poorly-fitting datapoints*
  - A change of priorities towards the peak
  - This assumes at least some observed data is good
- *Robust speech synthesis* is speech synthesis incorporating robust statistical techniques

Background section

Natural Speech Technology

Setting the stage:

1. Introduction
2. Sensitivity issues
3. Error sources
4. Proposed solutions

Some relevant proposals

Natural Speech Technology

Some TTS techniques for overcoming the mismatch in priorities:

1. Ensure high data quality
   - Example: Recording procedures and data cleaning scripts
2. Fit the peak of the model
   - Example: Minimum generation error training (MGE)
3. Ignore the tails of the data
   - Example: Component selection in mixture models (GMM-MDNs)
4. Do both 2 and 3!
   - β-estimation (Basu et al., 1998) rather than MLE
   - Example: In this talk!

Ensuring high data quality

Natural Speech Technology

Since TTS is so sensitive to data issues (Yamagishi et al., 2008), only use high-quality speech data to train systems:

+ Statistically robust in some cases
– Good data is expensive
  - Restricts synthesis to small and artificial datasets
– Does not address misspecification, only contamination

Exceedingly common, but seldom motivated through robustness

Minimum generation error

Find parameters that maximise the similarity between the mode $\widehat{\boldsymbol{x}}$ and the training data (MGE; Wu and Wang, 2006):

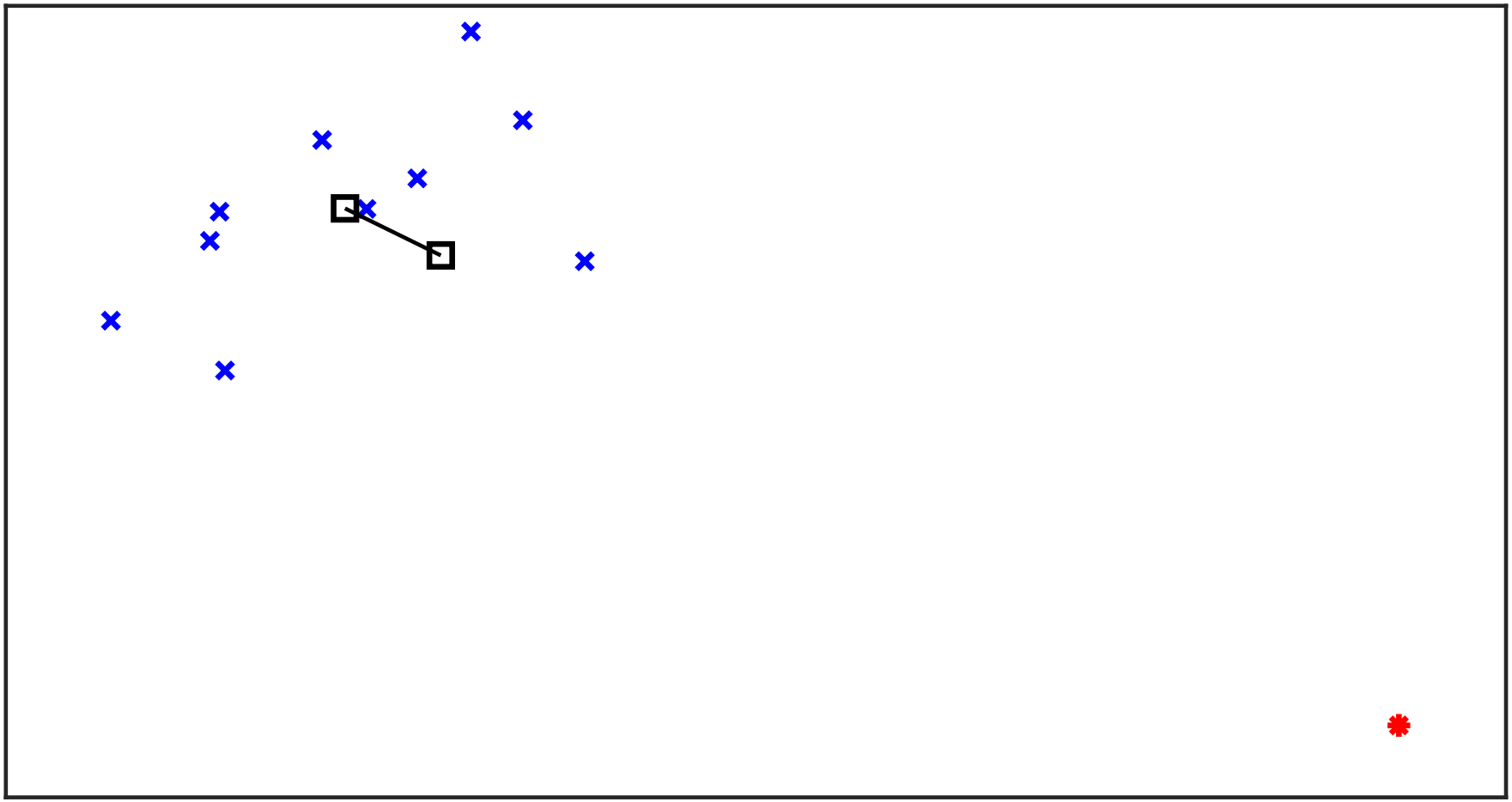
+ Explicitly optimises the mode
  - Only the mode matters at generation time, after all
– The MSE objective function is standard
  - Mathematically the same as MLE with a fixed-variance Gaussian
  - Not robust to contamination or misspecification

Not widely used

Minimum generation error training

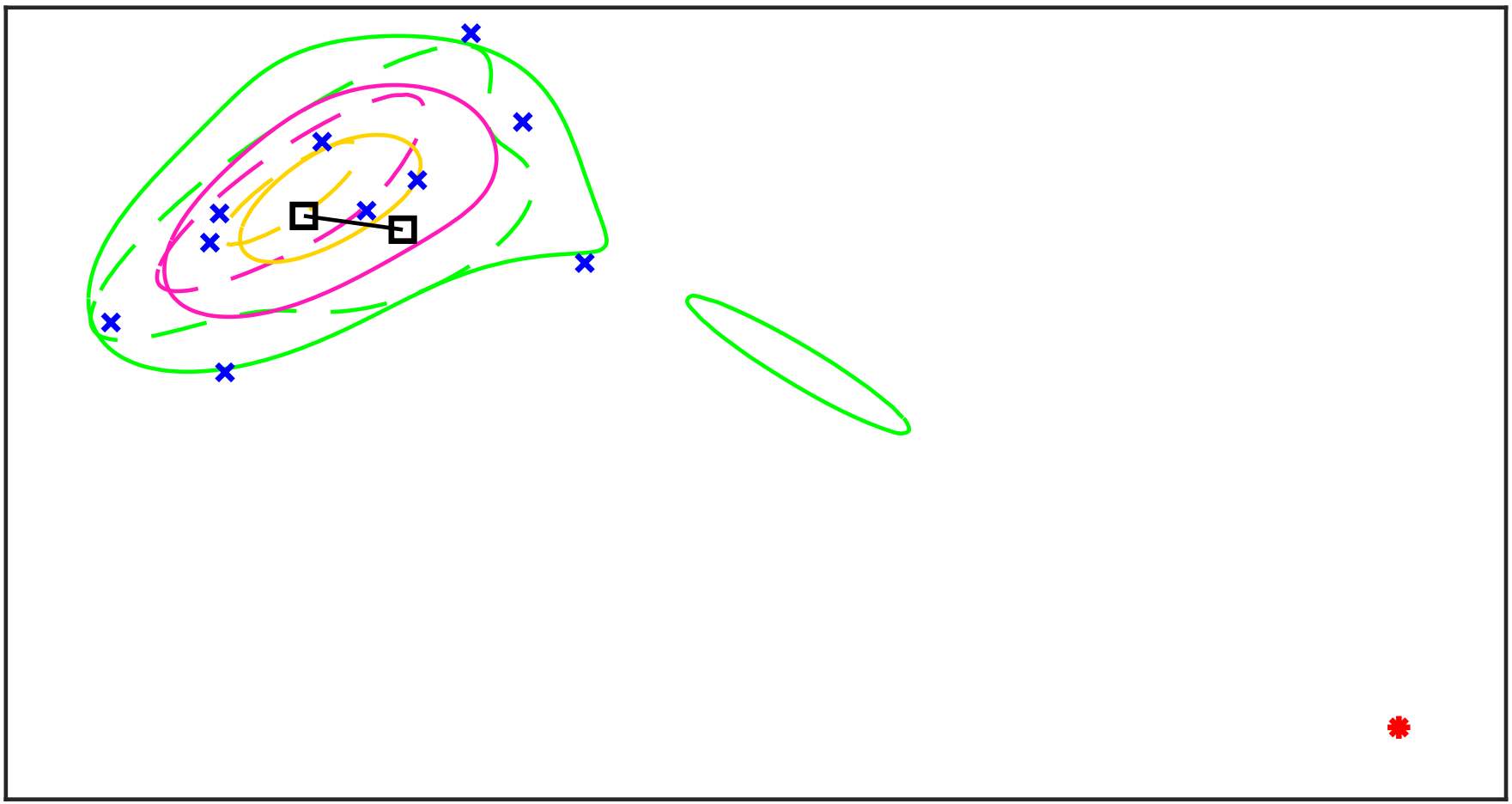
Natural Speech Technology

Minimum MSE MGE is just as fragile as Gaussian MLE!



Mixture models

Additional mixture components can absorb garbage datapoints



Component selection

Natural Speech Technology

Generate from the most massive mixture component:

+ Probabilistic
+ Greater modelling power than a single Gaussian
+ Can be statistically robust
– Different models for training and synthesis

While used in synthesis (Zen and Senior, 2014; Wang et al., 2016), it had previously not been motivated through robustness

A proposal

Natural Speech Technology

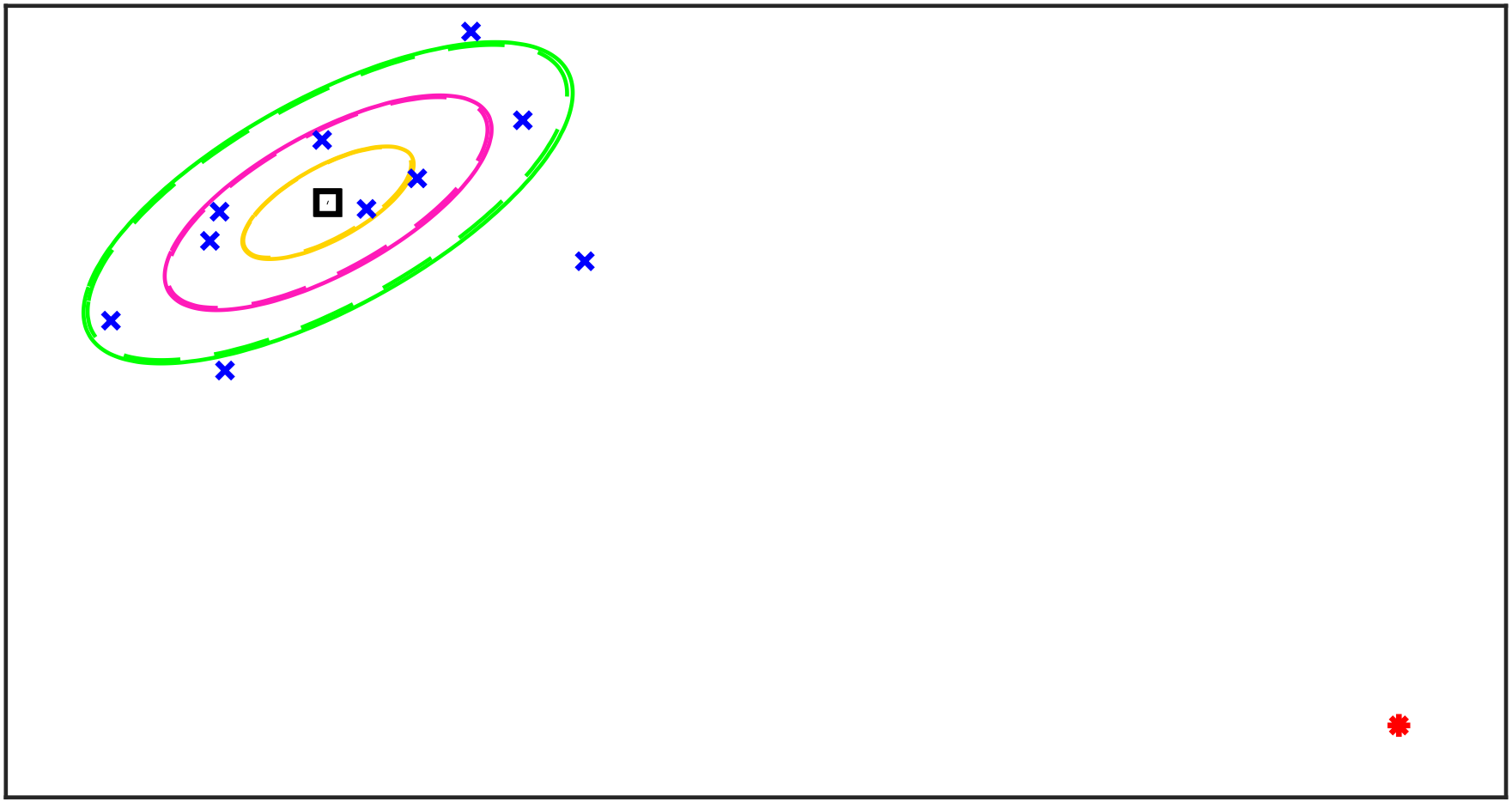
This talk describes a novel approach to the mismatch problem in data generation, applied to speech synthesis in Henter et al. (2016):

Use β-estimation, an alternative principle for parameter estimation that reduces the focus on low-probability regions, compared to MLE

+ Probabilistic
+ Robust
+ Directly addresses estimation-generation mismatch
+ Uses standard, Gaussian models

β-estimation example

Gaussian distribution fit using $\beta = 1/3$



Overview

1. Background
2. Theory
3. Application
4. Conclusion

Theory section

Natural Speech Technology

A closer look at the theory behind the mismatch problem, and a proposed solution:

1. Maximum likelihood estimation
2. The root of the problem
3. β-estimation
4. Properties of β-estimation

The scenario

1. Consider $\boldsymbol{X} \in \mathbb{R}^D$ distributed according to $f^\star(\boldsymbol{x})$
2. Let the *model* (parametric family) be specified by a pdf $f(\boldsymbol{x}; \boldsymbol{\theta})$ with parameter $\boldsymbol{\theta} \in \boldsymbol{\Theta}$
3. Let the training data $\mathcal{D} = \{\boldsymbol{x}_n\}_{n=1}^N$ be i.i.d. samples from $f^\star$
4. *Parameter estimation* is the task of finding $\widehat{\boldsymbol{\theta}}$ using $\mathcal{D}$ so that $f\left(\boldsymbol{x}; \widehat{\boldsymbol{\theta}}\right)$ approximates $f^\star(\boldsymbol{x})$ as well as possible

Maximum likelihood estimation

Natural Speech Technology

Maximum likelihood parameter estimation maximises the joint probability of the entire dataset:

$$\begin{aligned}\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\left(\mathcal{D}\right) &= \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \mathbb{P}\left(\mathcal{D};\, \boldsymbol{\theta}\right) \\ &= \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \prod_{n=1}^{N} f\left(\boldsymbol{x}_n;\, \boldsymbol{\theta}\right) \\ &= \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \sum_{n=1}^{N} \ln f\left(\boldsymbol{x}_n;\, \boldsymbol{\theta}\right)\end{aligned}$$

Clearly, the model must explain every datapoint simultaneously: if $f\left(\boldsymbol{x}_n;\, \boldsymbol{\theta}\right) = 0$ for any n, the likelihood is minimal

Why MLE?

Natural Speech Technology

Asymptotically, maximum likelihood estimation is:

1. Unbiased
2. Consistent
3. Efficient

...assuming clean data and no misspecification
($\exists\, \boldsymbol{\theta}^\star \in \boldsymbol{\Theta} : f^\star(\boldsymbol{x}) \equiv f(\boldsymbol{x}; \boldsymbol{\theta}^\star)$)

Gaussian MLE

The maximum-likelihood estimate of the Gaussian mean is the sample mean:

$$\widehat{\boldsymbol{\mu}}_{\mathrm{ML}}\left(\mathcal{D}\right) = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{x}_n$$

This is not robust: by changing only $\boldsymbol{x}_1$, $\widehat{\boldsymbol{\mu}}_{\mathrm{ML}}$ can be forced to equal any value

Theory section

Natural Speech Technology

A closer look at the theory behind the mismatch problem, and a proposed solution:

1. Maximum likelihood estimation
2. The root of the problem
3. β-estimation
4. Properties of β-estimation

The real issue with MLE

Natural Speech Technology

Theory will reveal the root of the problem:

1. Likelihood maximisation is KL-divergence minimisation (Akaike, 1973)
2. The KL-divergence focusses on the tails of the distribution

A simple fact

Write down the Kullback-Leibler divergence (KLD) between true distribution and model:

$$
\begin{aligned}
D_{\mathrm{KL}}\left(f^{\star} \,||\, f\right) &= \int f^{\star}\left(\boldsymbol{x}\right) \ln \frac{f^{\star}\left(\boldsymbol{x}\right)}{f\left(\boldsymbol{x};\, \boldsymbol{\theta}\right)} d\boldsymbol{x} \\
&= \int f^{\star}\left(\boldsymbol{x}\right) \ln f^{\star}\left(\boldsymbol{x}\right) d\boldsymbol{x} - \int f^{\star}\left(\boldsymbol{x}\right) \ln f\left(\boldsymbol{x};\, \boldsymbol{\theta}\right) d\boldsymbol{x} \\
&= h\left(f^{\star}\right) - \mathbb{E}_{f^{\star}}\left(\ln f\left(\boldsymbol{X};\, \boldsymbol{\theta}\right)\right)
\end{aligned}
$$

where $h\left(\cdot\right)$ is the differential entropy

A neat insight

Natural Speech Technology

Identify the parameter value that minimises the KLD:

$$\begin{aligned}\underset{\theta\in\Theta}{\operatorname{argmin}}\, D_{\mathrm{KL}}\left(f^{\star} \mid\mid f\right) &= \underset{\theta\in\Theta}{\operatorname{argmin}}\left(h\left(f^{\star}\right) - \mathbb{E}_{f^{\star}}\left(\ln f\left(\boldsymbol{X};\, \boldsymbol{\theta}\right)\right)\right) \\ &= \underset{\theta\in\Theta}{\operatorname{argmax}}\, \mathbb{E}_{f^{\star}}\left(\ln f\left(\boldsymbol{X};\, \boldsymbol{\theta}\right)\right) \\ &\approx \underset{\theta\in\Theta}{\operatorname{argmax}}\, \frac{1}{N}\sum_{n=1}^{N} \ln f\left(\boldsymbol{x}_n;\, \boldsymbol{\theta}\right)\end{aligned}$$

since $h\left(f^{\star}\right)$ is a constant w.r.t. $\boldsymbol{\theta}$ and the unknown expected value can be approximated by the sample mean over $\mathcal{D}$

In other words

Likelihood maximisation $\Leftrightarrow$ minimising the (empirical) KL-divergence:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ML}}\left(\mathcal{D}\right) = \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \frac{1}{N}\sum_{n=1}^{N} \ln f\left(\boldsymbol{x}_n;\, \boldsymbol{\theta}\right)$$
$$\rightarrow \operatorname*{argmin}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} D_{\mathrm{KL}}\left(f^{\star} \,||\, f\right)$$

as $N \rightarrow \infty$

- Distribution estimation, not parameter estimation
- Now we know where MLE converges under misspecification

MLE sensitivity explained

The KLD is highly sensitive to the tails of the data distribution:

1. The support of f must cover the support of $f^\star$:

$$\exists X_0 : \int_{X_0} f^\star(\boldsymbol{x})\, d\boldsymbol{x} > 0 = \int_{X_0} f(\boldsymbol{x}; \boldsymbol{\theta})\, d\boldsymbol{x} \Rightarrow D_{\mathrm{KL}}(f^\star \,||\, f) = \infty$$

   so all possible outcomes must be explained by the model

2. For small differences $f \approx f^\star$ the KLD is similar to a squared error weighted by $1/f^\star$ (Basu et al., 1998):

$$D_{\mathrm{KL}}(f^\star \,||\, f) \approx \int \frac{1}{f^\star(\boldsymbol{x})} \left(f^\star(\boldsymbol{x}) - f(\boldsymbol{x}; \boldsymbol{\theta})\right)^2 d\boldsymbol{x}$$

   - "The less probable, the more important"
   - The mode of $f^\star$ is given the lowest weight of all

Huh?

Let's take that again:

- *MLE gives the mode of $f^\star$ – the only point that matters for output generation – the lowest weight of all!*

This deep-seated estimation-generation mismatch has not previously been recognised in speech synthesis

- For classification, it is well known that accuracy matters most near class boundaries (and that MLE should be avoided)
- Not common knowledge in data-generation tasks

Theory section

Natural Speech Technology

A closer look at the theory behind the mismatch problem, and a proposed solution:

1. Maximum likelihood estimation
2. The root of the problem
3. β-estimation
4. Properties of β-estimation

A possible fix

Idea: Reduce the weight of the tails when fitting!

Try the β-divergences introduced by Basu et al. (1998) and Eguchi and Kano (2001):

$$D_\beta\left(f^\star \,||\, f\right) = \frac{1}{\beta}\int \left(f^\star\left(\boldsymbol{x}\right)\right)^{1+\beta} d\boldsymbol{x} + \int \left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{1+\beta} d\boldsymbol{x} - \frac{1+\beta}{\beta}\int f^\star\left(\boldsymbol{x}\right)\left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{\beta} d\boldsymbol{x}$$

where $\beta > 0$ is a tuning parameter

De-weighting the tails

When $f^\star(\boldsymbol{x}) \approx f$, we have

$$D_\beta\left(f^\star \,||\, f\right) \approx \frac{1+\beta}{2} \int \left(f^\star(\boldsymbol{x})\right)^{\beta-1} \left(f^\star(\boldsymbol{x}) - f\left(\boldsymbol{x};\, \boldsymbol{\theta}\right)\right)^2 d\boldsymbol{x}$$

so the difference in weight between peak(s) and tails decreases as β increases towards 1, as desired

Two special cases

One can show that

$$\lim_{\beta \to 0} D_\beta \left(f^\star \mid\mid f \right) = D_{\mathrm{KL}} \left(f^\star \mid\mid f \right)$$

$$D_1 \left(f^\star \mid\mid f \right) = \int \left(f^\star \left(\boldsymbol{x} \right) - f \left(\boldsymbol{x}; \boldsymbol{\theta} \right) \right)^2 d\boldsymbol{x}$$

so the β-divergences provide a continuum between the KL-divergence and complete tail-peak equality

Do the math

Natural Speech Technology

Similar to the KLD case, we work out the β-divergence between the two distributions

$$
\begin{aligned}
D_\beta\left(f^\star \,||\, f\right) &= \frac{1}{\beta}\int \left(f^\star\left(\boldsymbol{x}\right)\right)^{1+\beta} d\boldsymbol{x} + \int \left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{1+\beta} d\boldsymbol{x} \\
&\quad - \frac{1+\beta}{\beta}\int f^\star\left(\boldsymbol{x}\right)\left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{\beta} d\boldsymbol{x} \\
&= \frac{1}{\beta}\int \left(f^\star\left(\boldsymbol{x}\right)\right)^{1+\beta} d\boldsymbol{x} + \int \left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{1+\beta} d\boldsymbol{x} \\
&\quad - \frac{1+\beta}{\beta}\mathbb{E}_{f^\star}\left(\left(f\left(\boldsymbol{X};\boldsymbol{\theta}\right)\right)^{\beta}\right)
\end{aligned}
$$

β-estimation

Natural Speech Technology

Eliminate constants and substitute in the sample mean as before

$$
\begin{aligned}
&\operatorname*{argmin}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} D_\beta\left(f^\star \,||\, f\right) \\
&= \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \left(\mathbb{E}_{f^\star}\left(\left(f\left(\boldsymbol{X};\boldsymbol{\theta}\right)\right)^\beta\right) - \frac{\beta}{1+\beta}\int \left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{1+\beta} d\boldsymbol{x} \right) \\
&\approx \operatorname*{argmax}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \left(\frac{1}{N}\sum_{n=1}^{N}\left(f\left(\boldsymbol{x}_n;\boldsymbol{\theta}\right)\right)^\beta - \frac{\beta}{1+\beta}\int \left(f\left(\boldsymbol{x};\boldsymbol{\theta}\right)\right)^{1+\beta} d\boldsymbol{x} \right) \\
&= \widehat{\boldsymbol{\theta}}_{\mathrm{M}\beta}\left(\mathcal{D}\right)
\end{aligned}
$$

For lack of a better term, we will call this β-*estimation*

Theory section

A closer look at the theory behind the mismatch problem, and a proposed solution:

1. Maximum likelihood estimation
2. The root of the problem
3. β-estimation
4. Properties of β-estimation

Sums versus products

- In MLE, we maximise the sum of datapoint log-probabilities

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \prod_{n=1}^{N} f\left(\boldsymbol{x}_n; \boldsymbol{\theta}\right) = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{n=1}^{N} \ln f\left(\boldsymbol{x}_n; \boldsymbol{\theta}\right)$$

  which is dominated by the worst-fitting points
- In β-estimation, we maximise the sum of datapoint probabilities taken to a power

$$\underset{\boldsymbol{\theta}}{\operatorname{argmax}} \sum_{n=1}^{N} \left(f\left(\boldsymbol{x}_n; \boldsymbol{\theta}\right)\right)^{\beta}$$

  so a poorly-fitting point will only give a finite penalty

The Gaussian case

The β-estimate of the Gaussian mode is a weighted mean

$$\widehat{\boldsymbol{\mu}}_{\mathrm{M}\beta}\left(\mathcal{D}\right) = \sum_{n=1}^{N} \frac{f_{\mathcal{N}}\left(\boldsymbol{x}_n;\, \widehat{\boldsymbol{\mu}}_{\mathrm{M}\beta},\, \widehat{\boldsymbol{\Sigma}}_{\mathrm{M}\beta}\right)^{\beta}}{\sum_{n'=1}^{N} f_{\mathcal{N}}\left(\boldsymbol{x}_{n'};\, \widehat{\boldsymbol{\mu}}_{\mathrm{M}\beta},\, \widehat{\boldsymbol{\Sigma}}_{\mathrm{M}\beta}\right)^{\beta}} \boldsymbol{x}_n$$

- Weights rapidly go to zero for points away from $\widehat{\boldsymbol{\mu}}_{\mathrm{M}\beta}$
- Unlike the sample mean from MLE, bad points receive very small weight

Statistical properties

Natural Speech Technology

Basu et al. (1998) show that β-estimation is:

1. Consistent
   - If $\exists\, \boldsymbol{\theta}^\star \in \boldsymbol{\Theta} : f^\star(\boldsymbol{x}) \equiv f(\boldsymbol{x}; \boldsymbol{\theta}^\star)$
2. Robust
   - $\widehat{\boldsymbol{\theta}}_{\mathrm{M}\beta}(\mathcal{D})$ is a type of M-estimator (Huber, 2011)
3. Not maximally efficient
   - Since observations are discarded, more data is required to reach a certain estimation accuracy
   - The expected amount of data discarded can be used to set β, as a bias-variance trade-off

Overview

1. Background
2. Theory
3. Application
4. Conclusion

Collaborator credit

Natural Speech Technology

This study is joint work with

Srikanth Ronanki
Oliver Watts
Zhizheng Wu
Mirjam Wester
Simon King

from the University of Edinburgh

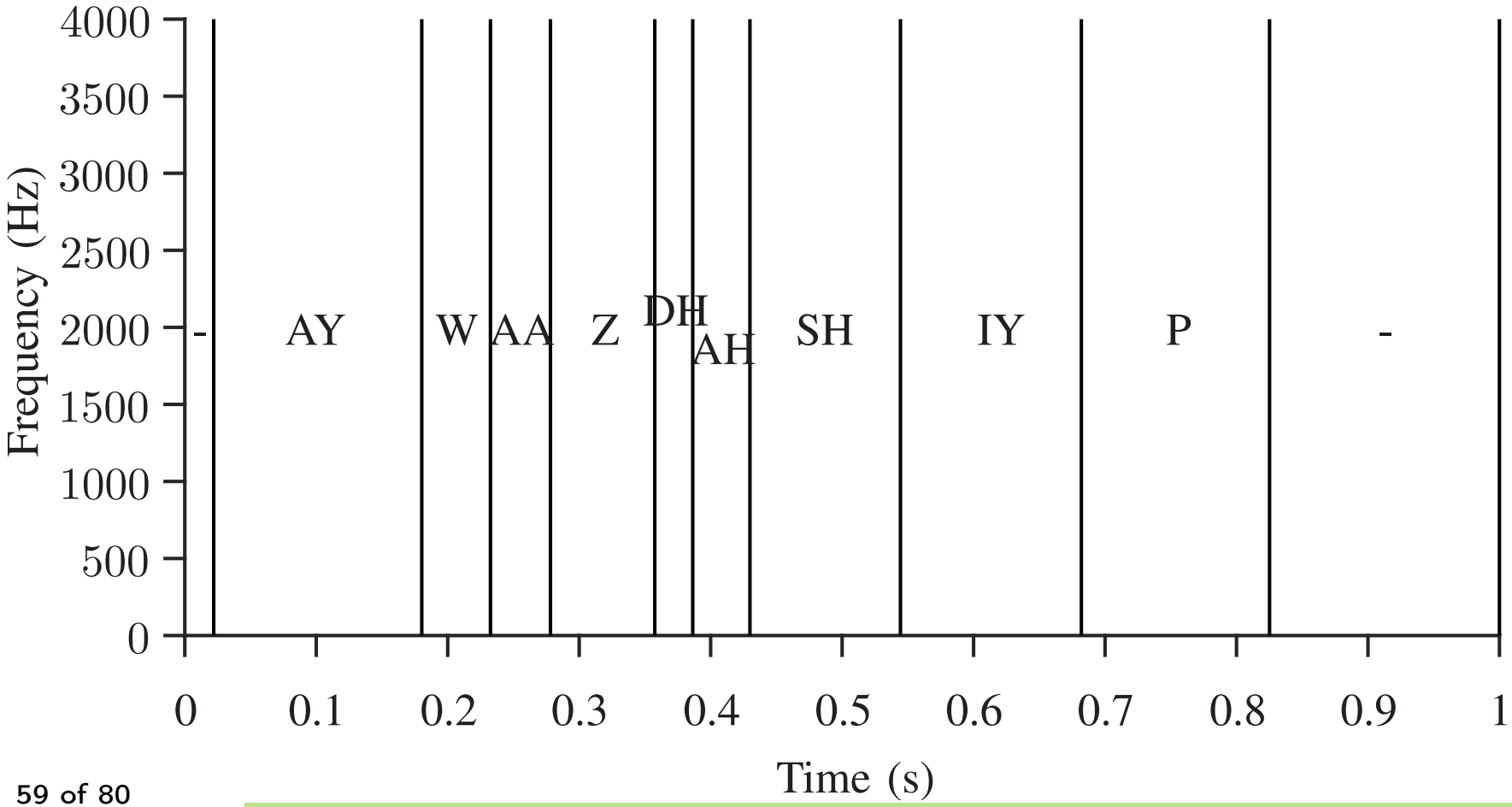
Application section

A first investigation into robust duration modelling for speech synthesis, reported in Henter et al. (2016):

1. Basic framework
2. Experiment setup
3. Results

TTS system overview

This application concentrates on the duration model



Robust duration modelling

Natural Speech Technology

- Why duration modelling?
  - Durations are important for sounding natural (stress, prosody)
  - Durations are hard to predict
  - Contrary to standard assumptions, durations are typically skewed and non-Gaussian
- Application to found data (audiobook)
  - Substantial transcription and alignment issues
- Phoneme-level robustness
  - Disregarding sub-state duration vectors on a per-phone basis

Some definitions

- p is a phone instance
- $\boldsymbol{l}_p$ is a vector of (input) linguistic features
- $\boldsymbol{D}_p \in \mathbb{R}^D$ is a vector of stochastic (output) sub-state durations
- $\boldsymbol{d}_p$ is an outcome of $\boldsymbol{D}_p$
- $\mathcal{D} = \{(\boldsymbol{l}_p, \boldsymbol{d}_p)\}$ is a training dataset

Mixture density network

Natural Speech Technology

Assume phone durations are independent and follow a GMM

$$f_{\boldsymbol{D}}\left(\boldsymbol{d};\, \boldsymbol{\theta}\right) = \sum_{k=1}^{K} \omega_k \cdot f_{\mathcal{N}}(\boldsymbol{d};\, \boldsymbol{\mu}_k,\, \mathrm{diag}(\boldsymbol{\sigma}_k^2))$$

- Distribution parameters $\boldsymbol{\theta} = \{\omega_k,\, \boldsymbol{\mu}_k,\, \boldsymbol{\sigma}_k^2\}_{k=1}^{K}$ depend on $\boldsymbol{l}$ via a deep neural network (DNN) $\boldsymbol{\theta}\left(\boldsymbol{l};\, \boldsymbol{W}\right)$ with weights $\boldsymbol{W}$
  - This is a *mixture density network* (MDN; Bishop, 1994)
  - Probabilistic regression with the functional form given by a DNN
- Setting $K = 1$ yields a conventional Gaussian duration model

Estimation and generation

Network weights are optimised to maximise the likelihood

$$\widehat{\boldsymbol{W}}_{\text{ML}}\left(\mathcal{D}\right) = \underset{\boldsymbol{W}}{\text{argmax}} \sum_{p \in \mathcal{D}} \ln f_{\boldsymbol{D}}(\boldsymbol{d}_p;\, \boldsymbol{\theta}(\boldsymbol{l}_p;\, \boldsymbol{W}))$$

Component selection is used for output (Zen and Senior, 2014; Wang et al., 2016)

$$k_{\text{max}}\left(\boldsymbol{l}\right) = \underset{k}{\text{argmax}}\, \omega_k(\boldsymbol{l};\, \widehat{\boldsymbol{W}})$$

$$\widehat{\boldsymbol{d}}\left(\boldsymbol{l}\right) = \boldsymbol{\mu}_{k_{\text{max}}(\boldsymbol{l})}(\boldsymbol{l};\, \widehat{\boldsymbol{W}})$$

- This reduces to standard generation when $K = 1$

Application section

A first investigation into robust duration modelling for speech synthesis, reported in Henter et al. (2016):

1. Basic framework
2. Experiment setup
3. Results

Setup in brief

- **Data:** Vol. 3 of Jane Austen's "Emma" ($\approx$ 3 hours)
  - Freely available at librivox.org/emma-by-jane-austen-solo
  - Home recording with imperfect transcriptions, so the (estimated) training-data durations are sometimes highly incorrect
- **DNN:** 6 tanh layers with MDN output
  - Merlin TTS (Wu et al., 2016) in Theano
  - Very simple to change from MLE to β-estimation

Reference systems

Natural Speech Technology

VOC Vocoded held-out natural speech (top line)

Same acoustic DNN, but different phone duration models:

FRC Synthesised speech with durations from VOC

BOT Always use the mean duration of each phone (bottom line)

MSE MMSE DNN (baseline)

MLE1 Gaussian, deep MDN maximising likelihood

Robust systems

Natural Speech Technology

MLE3 Three-component ($K = 3$), deep MDN with MLE
- Synthesis from the maximum-weight component

B75 Gaussian, deep MDN optimising β-divergence
- Set to include approximately 75% of datapoints (assuming data is Gaussian; this gives $\beta = 0.358$)

B50 Gaussian, deep MDN optimising β-divergence
- Set to include approximately 50% of datapoints (assuming data is Gaussian; this gives $\beta = 0.663$)

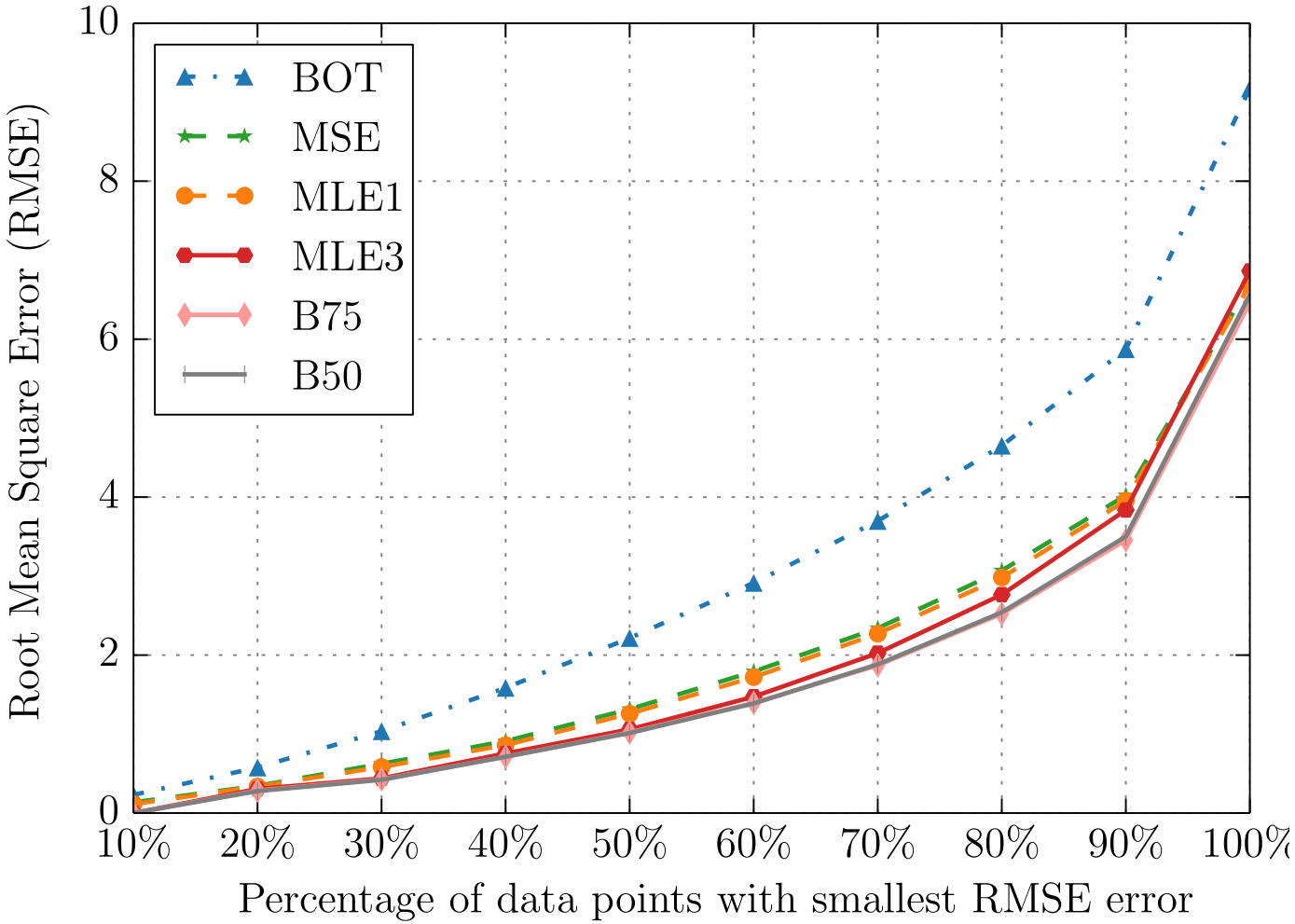
Application section

A first investigation into robust duration modelling for speech synthesis, reported in Henter et al. (2016):

1. Basic framework
2. Experiment setup
3. Results

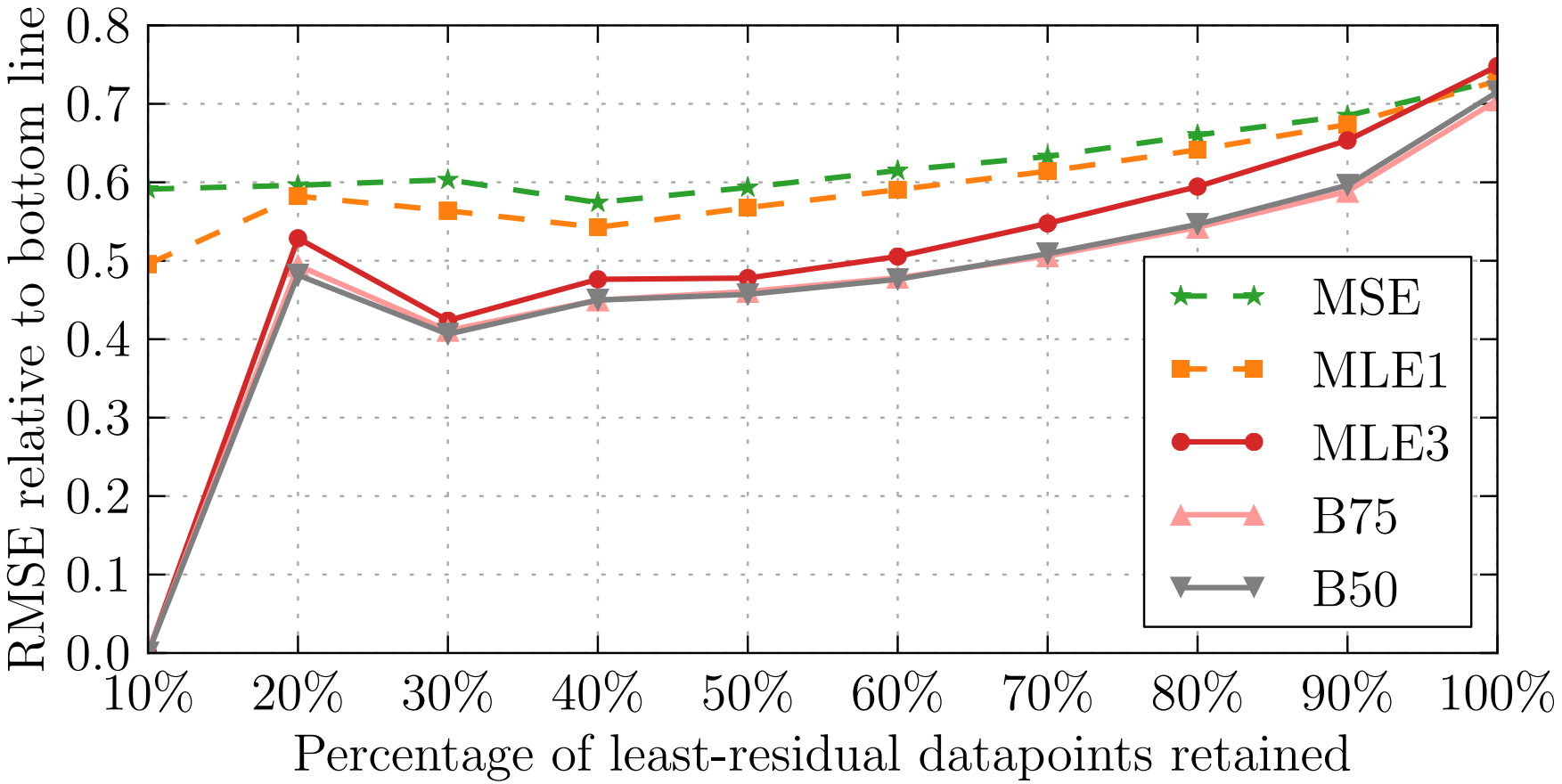
Outlier rejection

RMSE with respect to real durations (FRC) on test-data subsets:



Outlier rejection

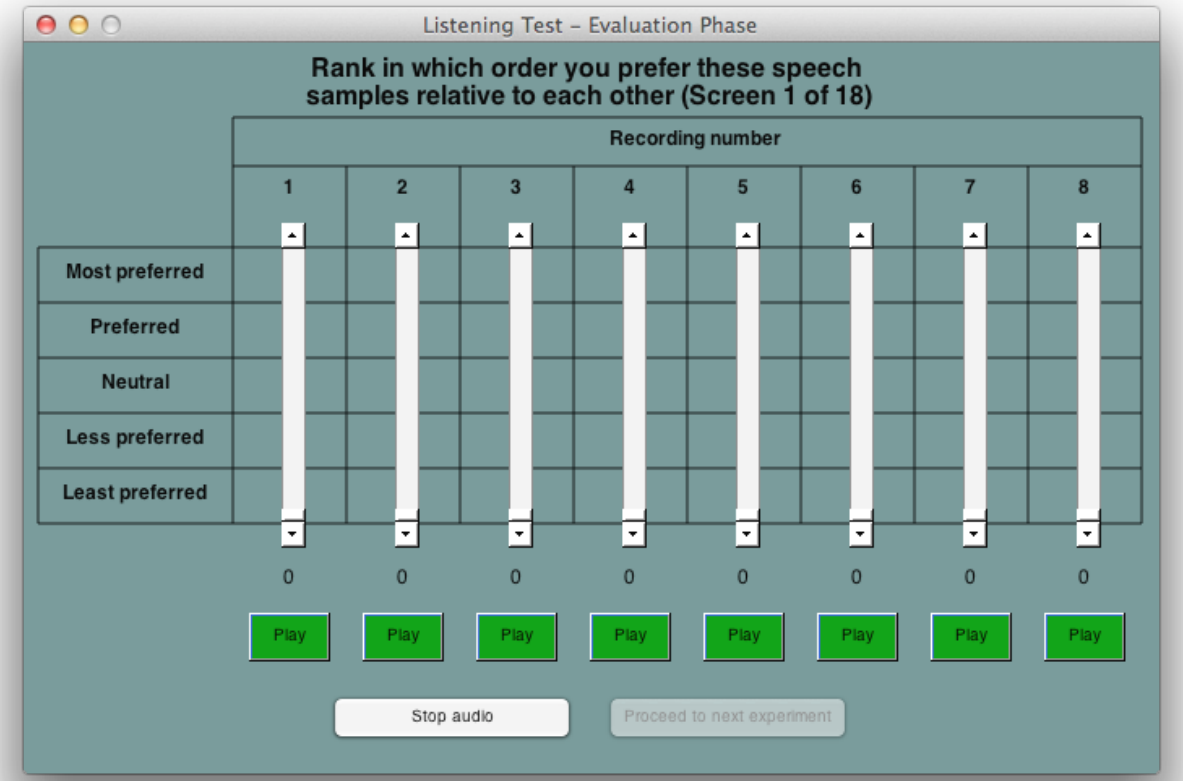
Relative RMSE on test-data subsets (with BOT at 1.0):



Subjective evaluation

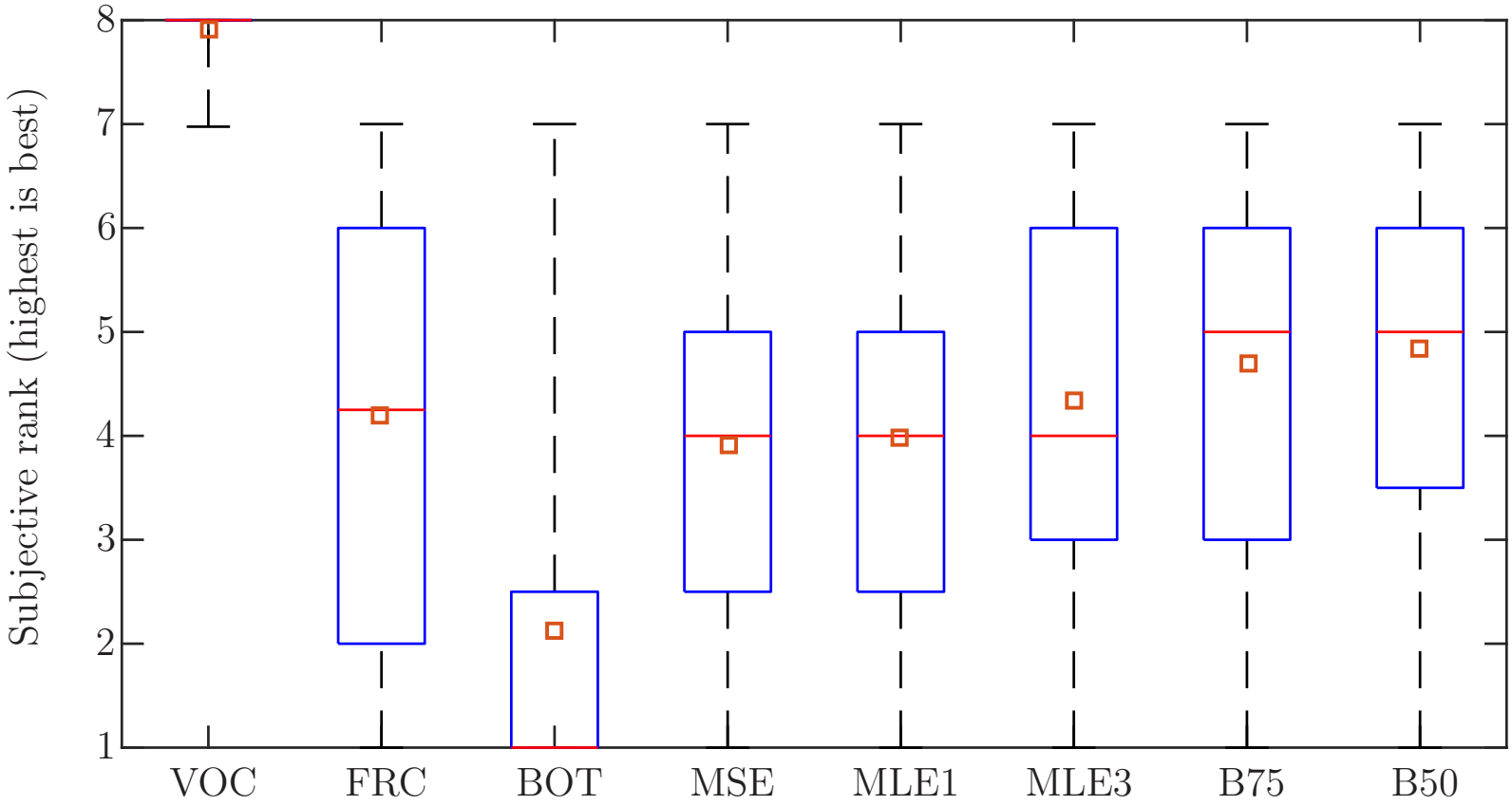
21 listeners ranked parallel examples of the different systems

- 21 sentences, of which each listener ranked 18



Subjective results

Listening test results, after converting to ranks:



Observations

- Robust duration models improve objective measures on the majority of the datapoints
  - Extreme examples are ignored, thus giving a better model of typical speech
- There are also improvements in subjective preference
  - Humans liked speech based on robust methods significantly more
- These advantages are not visible in objective error measures that are not themselves robust (e.g., MSE on entire dataset)
  - *Be careful about how you evaluate performance!*

Overview

1. Background
2. Theory
3. Application
4. Conclusion

Summary

1. Estimation and generation in data generation have fundamentally mismatched priorities
   - MLE cares most about the parts of the data that are the least relevant as output
   - This makes ML-estimated models highly sensitive to misspecification and bad training material
   - Minimum MSE-based prediction tasks are also affected

Summary

1. Estimation and generation in data generation have fundamentally mismatched priorities
   - MLE cares most about the parts of the data that are the least relevant as output
   - This makes ML-estimated models highly sensitive to misspecification and bad training material
   - Minimum MSE-based prediction tasks are also affected
2. Robust statistics offer a solution to the problems
   - A good fit for mode-based output generation
   - β-estimation is a simple method that directly addresses the estimation-generation mismatch
   - Robustness provides improved objective and subjective results

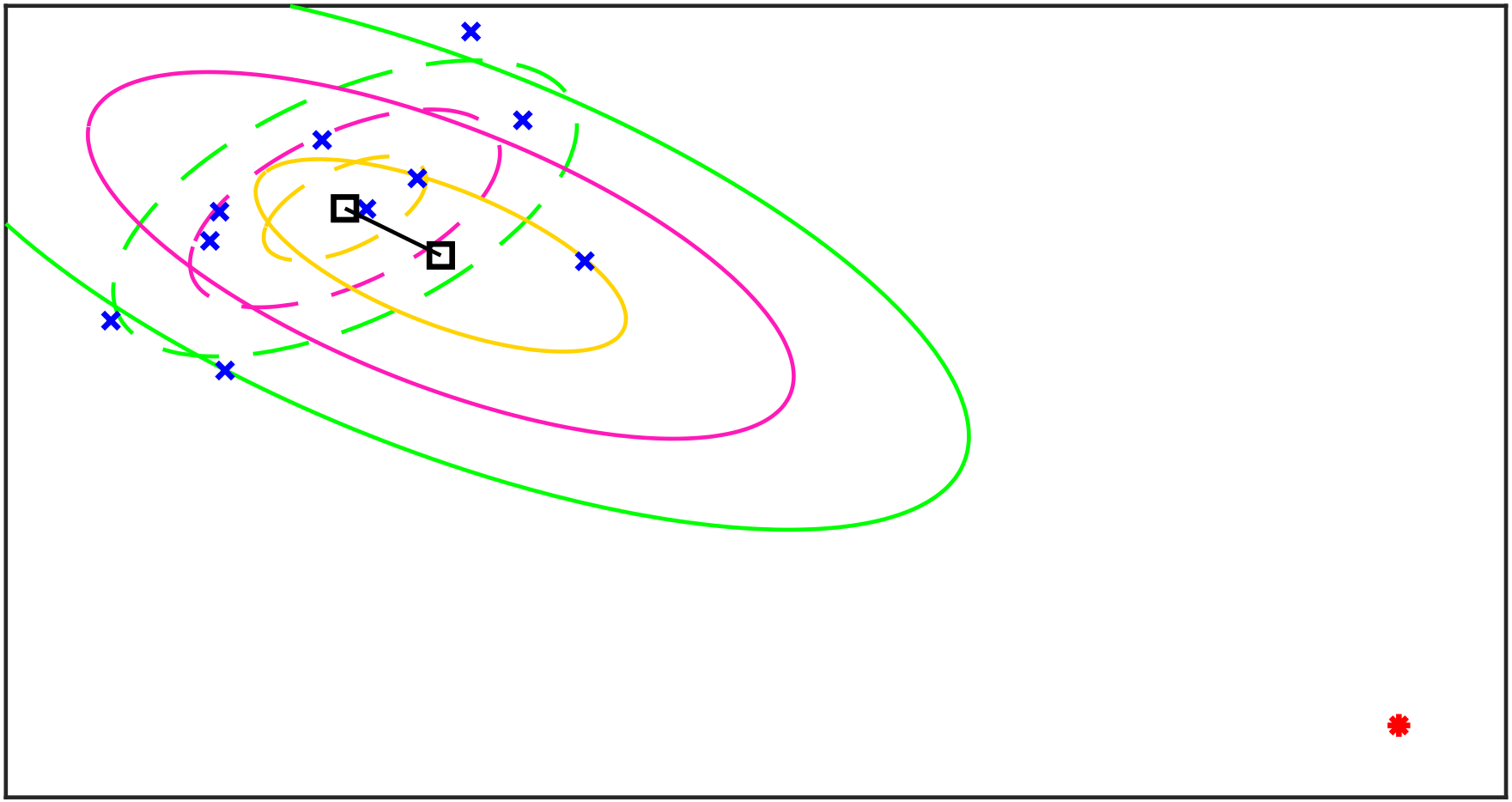
An aside on sampling

The KL-divergence (i.e., MLE) is inappropriate both for:

- Most probable output generation (Henter et al., 2016)
- Sampling from the fitted model (Theis et al., 2016)

Sampling is also affected

MLE places a lot of probablity mass in low-probability regions



Generative adversarial networks

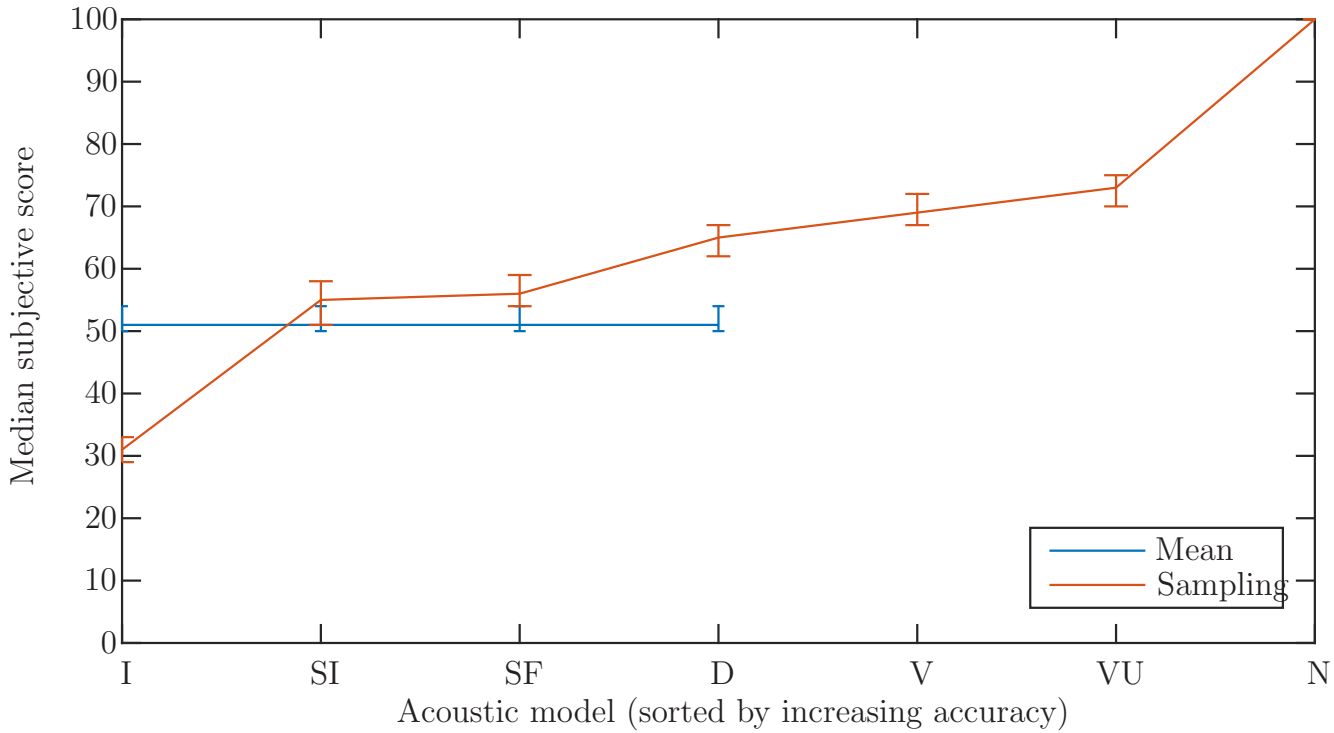
Natural Speech Technology

Generative adversarial networks (GANs; Goodfellow et al., 2014):

- State-of-the art for sampling from models in many tasks
- Represent "learned perception"
  - Models are scored based on distinguishability between real and synthetic examples
- Change the optimisation procedure, not necessarily its goal
  - Can optimise likelihood (Goodfellow, 2014), or robust error measures like Jensen-Shannon divergence (Goodfellow et al., 2014) or Wasserstein distance (Arjovsky et al., 2017)
  - Robustness is not inherent, but still a choice we have to make

Random or deterministic output?

Henter et al. (2014) found that the preferred generation strategy may depend on model accuracy:



- Only samples can become indistinguishable from real examples

Outlook

β-estimation and robust data generation methods fill an empty spot beside our best classification approaches:

Task	**Classification** (return label)	**Data generation** (return observation)	
Output	**Peak**	**Peak**	**Random**
Approach	**Instantiation/example**		

Outlook

β-estimation and robust data generation methods fill an empty spot beside our best classification approaches:

Task	**Classification** (return label)	**Data generation** (return observation)	
Output	**Peak**	**Peak**	**Random**
Approach	**Instantiation/example**		
Generic			
Task-oriented			
+ Robust			

Outlook

β-estimation and robust data generation methods fill an empty spot beside our best classification approaches:

Task	**Classification** (return label)	**Data generation** (return observation)	
Output	**Peak**	**Peak**	**Random**
Approach	**Instantiation/example**		
Generic	Generative MLE		
Task-oriented	MMI (cond. MLE)		
+ Robust	MWE/MPE/MCE (min. class. err.)		

Outlook

β-estimation and robust data generation methods fill an empty spot beside our best classification approaches:

Task	**Classification** (return label)	**Data generation** (return observation)	
Output	**Peak**	**Peak**	**Random**
Approach	**Instantiation/example**		
Generic	Generative MLE		
Task-oriented	MMI (cond. MLE)	MGE (MMSE)	MLE-GAN
+ Robust	MWE/MPE/MCE (min. class. err.)		

Outlook

β-estimation and robust data generation methods fill an empty spot beside our best classification approaches:

Task	**Classification** (return label)	**Data generation** (return observation)	
Output	**Peak**	**Peak**	**Random**
Approach	**Instantiation/example**		
Generic	Generative MLE		
Task-oriented	MMI (cond. MLE)	MGE (MMSE)	MLE-GAN
+ Robust	MWE/MPE/MCE (min. class. err.)	β-estimation Student's-t	JSD-GAN WGAN

The end

The end

Thank you for listening!

References (1)

Akaike, H. (1973). Information theory and an extension of the maximum likelihood principle. In *Proc. Second Int. Symp. Inf. Theory*, pages 267–281.

Aravkin, A. Y., van Leeuwen, T., and Herrmann, F. J. (2011). Robust full-waveform inversion using the Student's t-distribution. In *SEG Tech. Program Expand. Abstr.*, pages 2669–2673.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *Proc. ICML*.

Basu, A., Harris, I. R., Hjort, N. L., and Jones, M. C. (1998). Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559.

Bishop, C. M. (1994). Mixture density networks. Technical Report NCRG/94/004, Neural Computing Research Group, Aston University.

Domingos, P. (2000). A unified bias-variance decomposition for zero-one and squared loss. In *Proc. AAAI*, pages 564–569.

Eguchi, S. and Kano, Y. (2001). Robustifying maximum likelihood estimation. Technical Report Research Memo 802, Institute of Statistical Mathematics, Tokyo, Japan.

References (2)

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Proc. NIPS*, pages 2672–2680.

Goodfellow, I. J. (2014). On distinguishability criteria for estimating generative models. In *Proc. ICLR Workshop Track*.

Henter, G. E., Merritt, T., Shannon, M., Mayo, C., and King, S. (2014). Measuring the perceptual effects of modelling assumptions in speech synthesis using stimuli constructed from repeated natural speech. In *Proc. Interspeech*, pages 1504–1508.

Henter, G. E., Ronanki, S., Watts, O., Wester, M., Wu, Z., and King, S. (2016). Robust TTS duration modelling using DNNs. In *Proc. ICASSP*, pages 5130–5134.

Huber, P. J. (2011). *Robust Statistics*. Springer, New York, NY, 2nd edition.

Theis, L., van den Oord, A., and Bethge, M. (2016). A note on the evaluation of generative models. In *Proc. ICLR*.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T., and Kitamura, T. (2000). Speech parameter generation algorithms for HMM-based speech synthesis. In *Proc. ICASSP*, volume 3, pages 1315–1318.

References (3)

Uría, B., Murray, I., Renals, S., and Valentini-Botinhao, C. (2015). Modelling acoustic feature dependencies with artificial neural networks: Trajectory-RNADE. In *Proc. ICASSP*, pages 4465–4469.

Wang, W., Xu, S., and Xu, B. (2016). Gating recurrent mixture density networks for acoustic modeling in statistical parametric speech synthesis. In *Proc. ICASSP*, pages 5520–5524.

Wu, Y.-J. and Wang, R.-H. (2006). Minimum generation error training for HMM-based speech synthesis. In *Proc. ICASSP*, pages I–89–I–92.

Wu, Z., Watts, O., and King, S. (2016). Merlin: An open source neural network speech synthesis system. In *Proc. SSW*, pages 218–223.

Yamagishi, J., Ling, Z.-H., and King, S. (2008). Robustness of HMM-based speech synthesis. In *Proc. Interspeech*, pages 581–584.

Zen, H., Agiomyrgiannakis, Y., Egberts, N., Henderson, F., and Szczepaniak, P. (2016). Fast, compact, and high quality LSTM-RNN based statistical parametric speech synthesizers for mobile devices. In *Proc. Interspeech*, pages 2273–2277.

References (4)

Zen, H., Nose, T., Yamagishi, J., Sako, S., Masuko, T., Black, A., and Tokuda, K. (2007). The HMM-based speech synthesis system (HTS) version 2.0. In *Proc. SSW6*, pages 294–299.

Zen, H. and Senior, A. (2014). Deep mixture density networks for acoustic modeling in statistical parametric speech synthesis. In *Proc. ICASSP*, pages 3844–3848.

Example audio

Natural Speech Technology

Example utterance from held-out audiobook chapter:

VOC FRC BOT MSE MLE1
MLE3 B75 B50

Paths to robustness

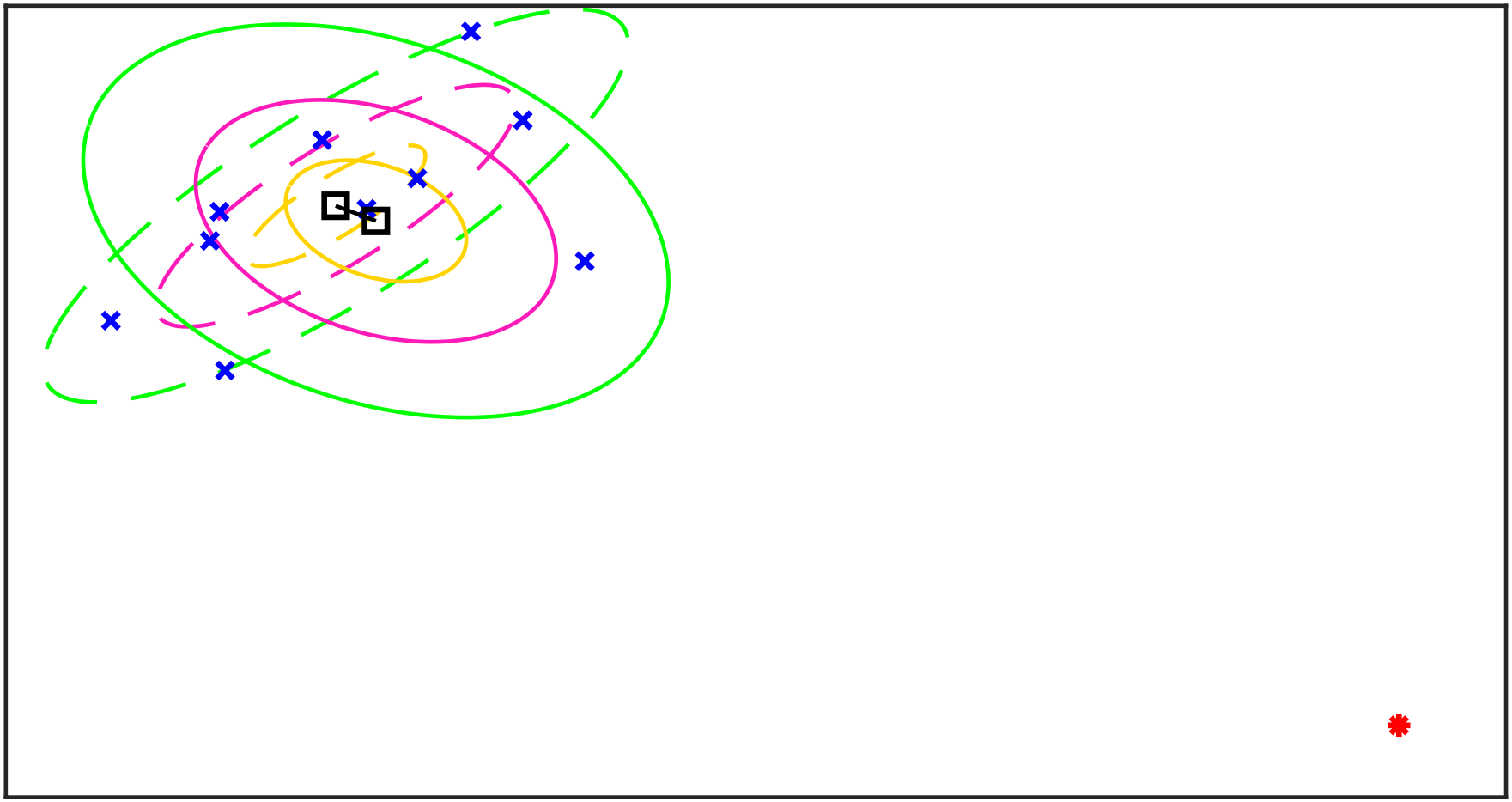
Natural Speech Technology

1. Clean or improve the data (everyone does this already)
2. Change the model family
   - Example: Richter distribution (Zen et al., 2016), Student's t-distribution (Aravkin et al., 2011)
3. Change the fitting principle
   - Example: β-estimation (Henter et al., 2016), non-MLE GANs (Goodfellow et al., 2014; Arjovsky et al., 2017)
4. Change the fitted model before generation
   - Example: Discard mixture components (Zen and Senior, 2014)

Fitting a fat-tailed distribution

Natural Speech Technology

Student's t-distribution with $\nu = 2.5$ fit using MLE



Preventing singularities

- Unlike MLE, β-estimation incorporates an explicit data-independent *concentration penalty*

$$I_\beta(\boldsymbol{\theta}) = \frac{\beta}{1+\beta} \int \left(f(\boldsymbol{x}; \boldsymbol{\theta})\right)^{1+\beta} d\boldsymbol{x}$$

  that prevents the estimation from explaining only a single datapoint
- If $f_{\mathcal{N}}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the Gaussian pdf, the concentration penalty can be computed from

$$\int \left(f_{\mathcal{N}}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})\right)^{1+\beta} d\boldsymbol{x} = (2\pi)^{-\frac{D\beta}{2}} (1+\beta)^{-\frac{D}{2}} (\det \boldsymbol{\Sigma})^{-\frac{\beta}{2}}$$

Effective sample size

If the Gaussian model is correct, how much more data will the robust procedure need to reach the same accuracy as MLE?

Results from Basu et al. (1998) give the effective sample size

$$N_{\text{eff}} \to \left(1 + \frac{\beta^2}{1+2\beta}\right)^{-\frac{D+2}{2}} N$$

- This gives an impression of the number of datapoints ignored
- If the data is not a perfect fit, more points are likely to be discarded

The robust trade-off

Natural Speech Technology

Robustness versus estimation accuracy is a kind of bias-variance (Domingos, 2000) trade-off:

- β too small
  - Sensitivity to bad data and bad assumptions (large bias)
- β too large
  - Sensitivity to random variation (large variance)
  - Uncommon behaviour might be ignored
    - Difficult-to-predict speech sounds might not be modelled
    - Another type of modelling bias

Tuning β

Natural Speech Technology

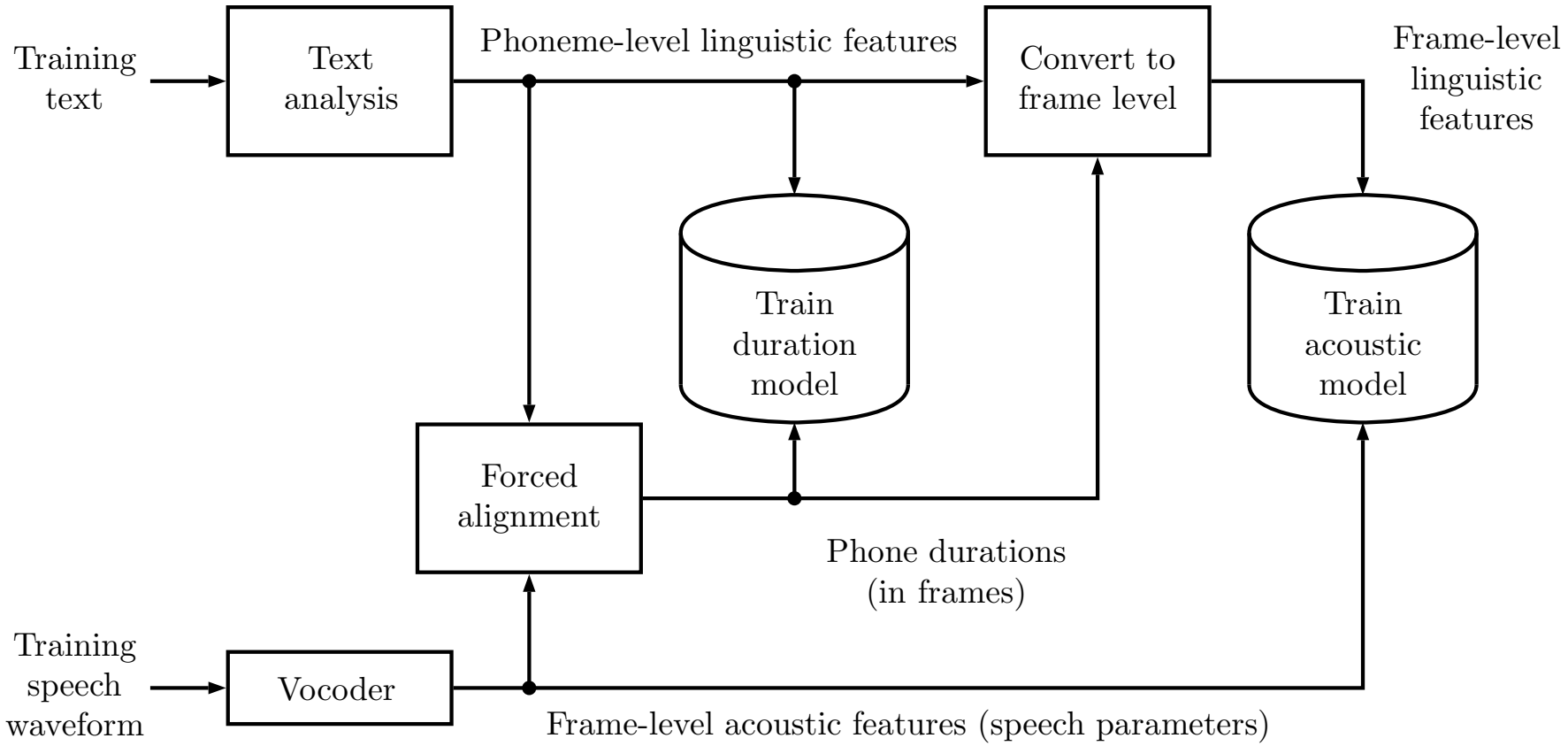
Estimating tuning parameters from data – including the necessary degree of robustness – is often not robust

We propose to choose β based on a certain target N_{eff}, say 0.8, with case-by-case modifications:

- The greater the data-model mismatch, the lower N_{eff}
  - In difficult situations, we need to be more robust
- With more data, N_{eff} can probably be reduced
  - Likely to give a lower bias for the same variance

TTS system overview

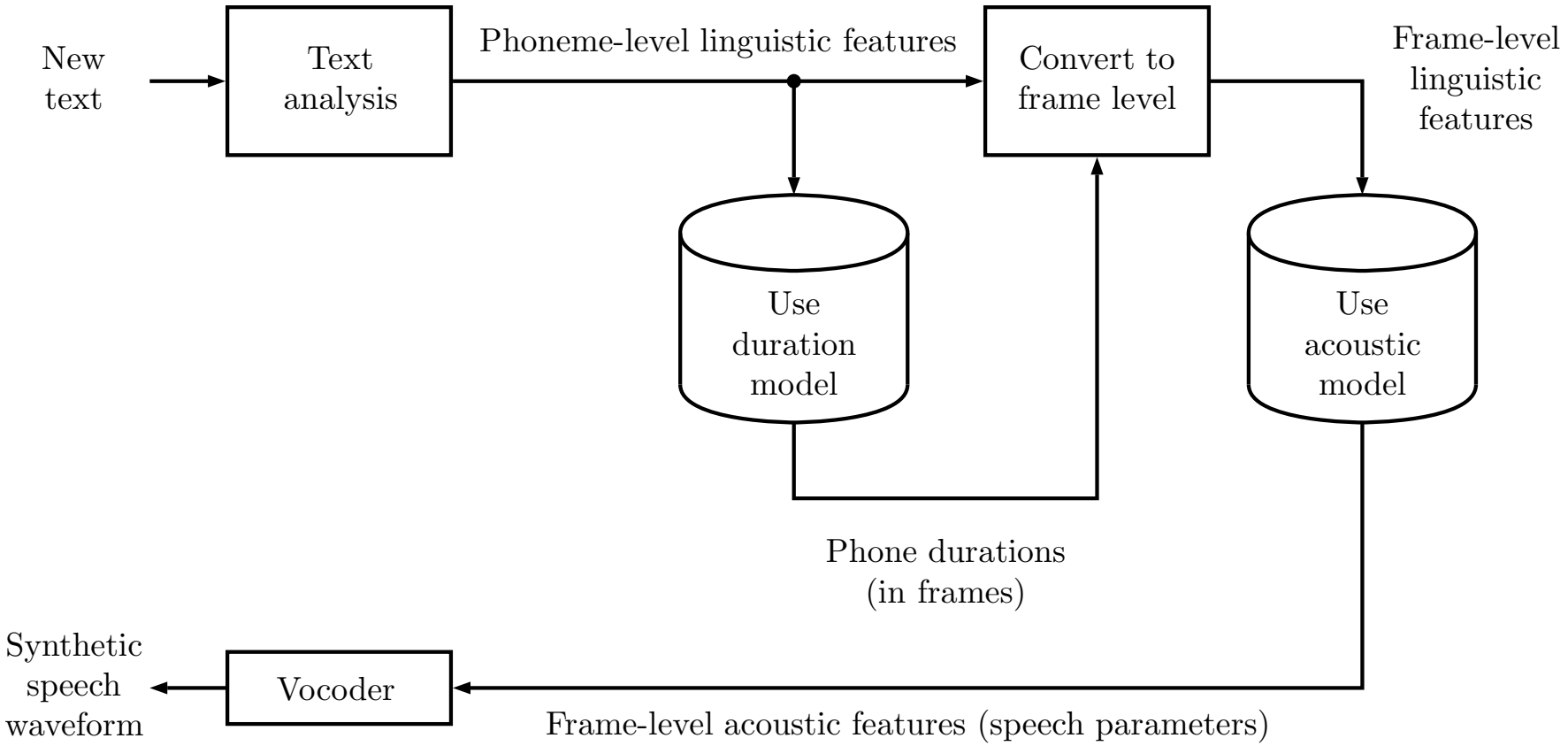
Training: Build models using parallel text and audio



TTS system overview

Natural Speech Technology

Synthesis: Use models to generate audio from input text



Data

Natural Speech Technology

Audiobooks are a classic source of found TTS data

- Jane Austen's "Emma" from LibriVox
  - Volume 3, chapters 1–10
  - Read by Sherry Crowther (US English)
- 1739 utterances (92,025 non-silent phones)
  - 175 minutes total, 6.06 s average utterance duration
  - Train/dev/test sets: 1660/39/40 utterances

Input and output features

- 200 frames per second at 44.1 kHz
- Linguistic features
  - Based on Festvox
  - One-hot encoding of 592 categorical features $\boldsymbol{l}^{(b)}$
  - Nine continuous-valued features $\boldsymbol{l}^{(d)}$, normalised to range [0.01, 0.99]
- Acoustic features $\boldsymbol{x}$
  - STRAIGHT vocoder
    - Log-F0, 60 spectrum mel-ceps, 25 baps
  - Statics, deltas, and delta-deltas ($\approx$ 250 dimensions total)
  - Each dimension normalised to zero mean and unit variance

Synthesis steps

1. `ehmm` for acoustics-based pause/silence insertion
   - Oracle pausing strategy
2. text & pausing information $\rightarrow$ binary linguistic features $\boldsymbol{l}^{(b)}$
3. $\boldsymbol{l}^{(b)} \rightarrow$ DNN-predicted per-phone (rounded) Gaussian mean state durations $\boldsymbol{d}$
4. $\boldsymbol{d} \rightarrow$ duration-based linguistic features $\boldsymbol{l}^{(d)}$
5. $\boldsymbol{l}^{(b)}$ & $\boldsymbol{l}^{(d)} \rightarrow$ DNN-predicted per-frame static & dynamic feature distributions
6. MLPG with postfiltering to generate acoustic parameter trajectories

Neural network design

- 6 hidden layers
  - 256/1024 units each (duration/acoustic model)
  - tanh activation function
- MDN parameter output layer
  - Softmax outputs for weights
  - Linear outputs for means
  - Logarithmic outputs with variance flooring for diagonal covariances

Implementation

Deep MDN code based on Wu et al. (2016)

- Setup largely follows Zen and Senior (2014)
  - Random initialisation
  - Trained until development set likelihood peaked
- GPU implementation with Python + Theano
  - Batched stochastic gradient descent
  - β-estimation straightforward to implement
    - Trained as refinements of less robust models (e.g., MLE)
  - Log-sum-exp trick for safe GMM likelihood evaluation

Listening test details

Natural Speech Technology

- 21 held-out sentences (2–8 seconds long) used
- MUSHRA/preference test hybrid
  - Stimuli presented in parallel (unlabelled, random order)
  - No designated reference stimulus
  - Instructed to rank the different stimuli by preference
- 21 listeners
  - Each ranked 18 sentences in a balanced design
  - Remaining sentences used for training and GUI tutorial

Minimum MSE is MLE is the mean

For a D-dimensional isotropic Gaussian distribution

$$f_{\mathcal{N}}(\boldsymbol{x}_n;\, \boldsymbol{\mu},\, \sigma^2 \boldsymbol{I}) = \left(2\pi\sigma^2\right)^{-\frac{D}{2}} e^{-\frac{1}{2}(\boldsymbol{x}_n-\boldsymbol{\mu})^{\mathsf{T}}\left(\sigma^2 \boldsymbol{I}\right)^{-1}(\boldsymbol{x}_n-\boldsymbol{\mu})}$$

the maximum-likelihood estimate of the location parameter $\boldsymbol{\mu}$ is

$$\begin{aligned}
\widehat{\boldsymbol{\mu}}_{\mathsf{ML}}\left(\mathcal{D}\right) &= \operatorname*{argmax}_{\boldsymbol{\mu}\in\mathbb{R}^D} \sum_{n=1}^{N} \ln f_{\mathcal{N}}(\boldsymbol{x}_n;\, \boldsymbol{\mu},\, \sigma^2 \boldsymbol{I}) && \text{(MLE of } \boldsymbol{\mu}\text{)} \\
&= \operatorname*{argmax}_{\boldsymbol{\mu}\in\mathbb{R}^D} \sum_{n=1}^{N} -\frac{1}{2\sigma^2}\left(\boldsymbol{x}_n-\boldsymbol{\mu}\right)^{\mathsf{T}}\left(\boldsymbol{x}_n-\boldsymbol{\mu}\right) \\
&= \operatorname*{argmin}_{\boldsymbol{\mu}\in\mathbb{R}^D} \sum_{n=1}^{N} \left\|\boldsymbol{x}_n-\boldsymbol{\mu}\right\|_2^2 && \text{(MMSE for } \boldsymbol{\mu}\text{)} \\
&= \tfrac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n = \overline{\boldsymbol{x}} && \text{(sample mean)}
\end{aligned}$$