STATISTICAL MODELLING: Theory and practice

Project 3: Financial data

GOALS: ASSIGNMENT 1

- Present the data
- Fit and asses **normal model**
- Present a **new hypothetically better model**
- Discuss which model is better

The financial data set

Weekly returns from Exchange Traded Fund (EFT)

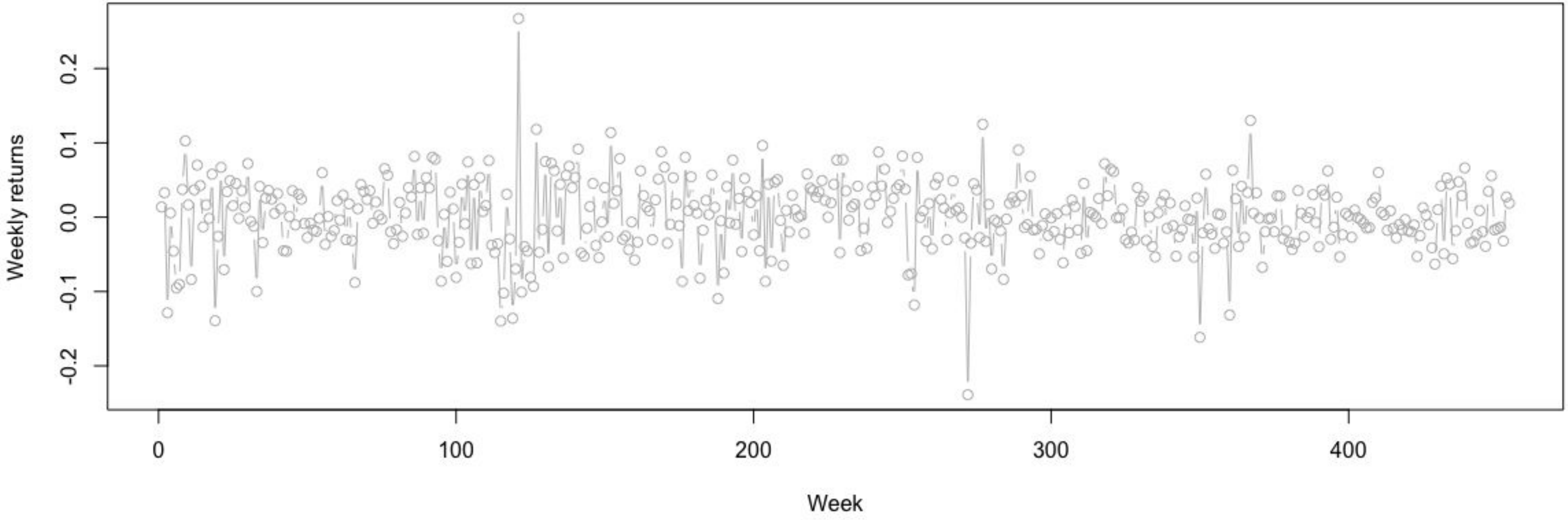
$$weekly\ returns = \frac{final\ price}{initial\ price} - 1$$

Data set

	time	SLV
1	2006-5-5	0.01376
2	2006-5-12	0.03286
3	2006-5-19	-0.12863
...	...	...
452	2015-4-24	-0.03213
453	2015-5-1	0.02722
454	2015-5-8	0.01875

Summary statistics of weekly returns

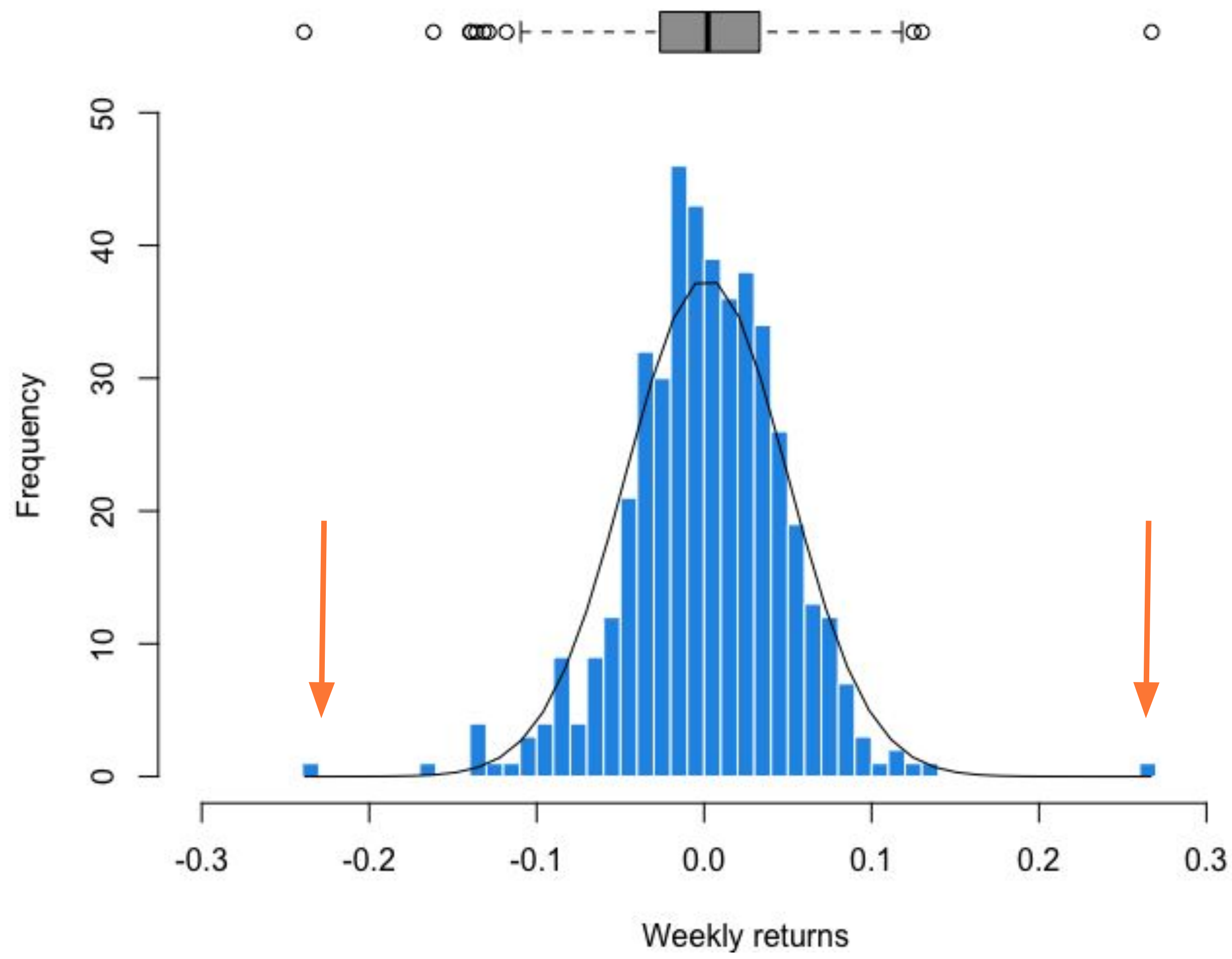
	SLV
Min. :	-0.238893
1st Qu.:	-0.026350
Median :	0.002226
Mean :	0.001468
3rd Qu.:	0.033122
Max. :	0.267308

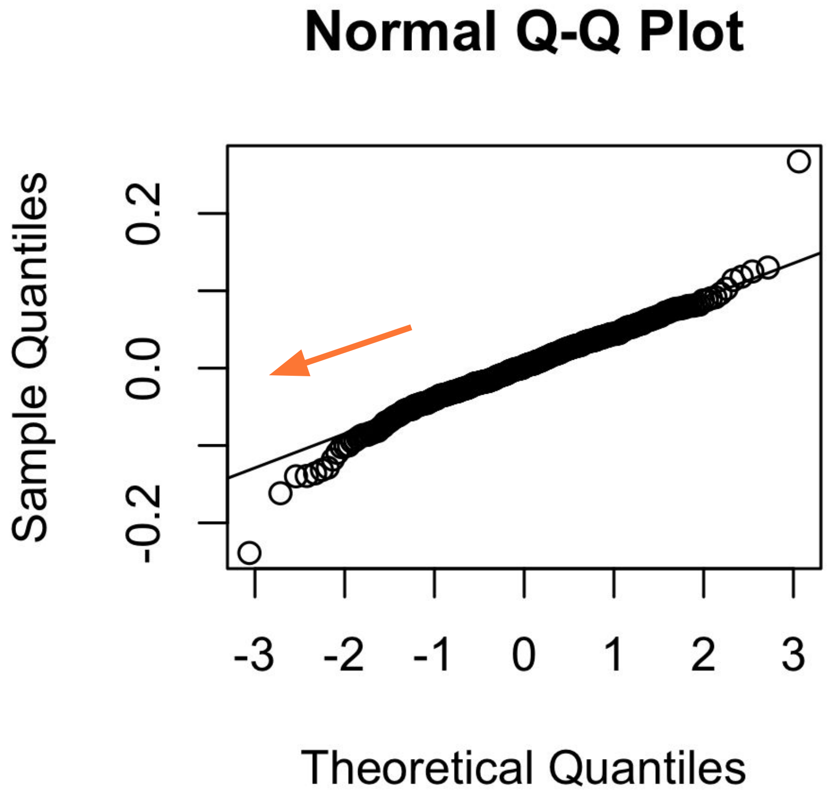


Fit to normal distribution

$$f_0(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

$$\log L(\mu, \sigma^2) = -\frac{n}{2} \log \sigma^2 - \frac{1}{2\sigma^2} \sum_i (x_i - \mu)^2,$$





Normal distribution

$$f_0(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

$$\log L(\mu, \sigma^2) = -\frac{n}{2}\log\sigma^2 - \frac{1}{2\sigma^2}\sum_i (x_i - \mu)^2,$$

Normal Model

	Est.	2.5%	97.5%
mu	0.001467	-0.00297	0.00591
sigma	0.04830	0.04385	0.05274

New model hypothesis: **Cauchy**

Cauchy distribution for heavy tails

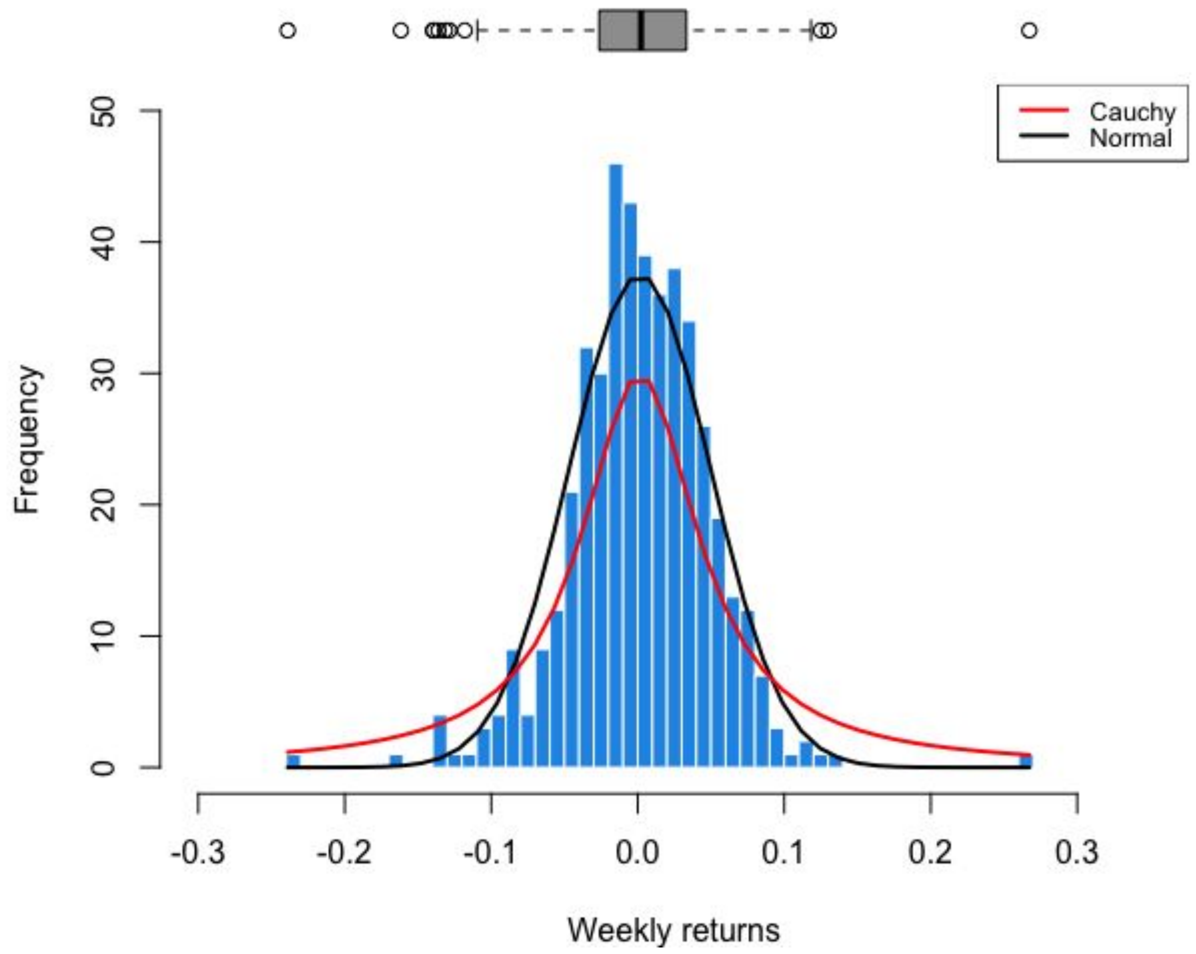
$$f_0(x) = \frac{1}{\pi(1 + x^2)}$$

$$L(\mu, \sigma) = \prod_i \frac{1}{\sigma}\left\{1 + \frac{(x_i - \mu)^2}{\sigma^2}\right\}^{-1}$$

Cauchy Model

	Est.	2.5%	97.5%
mu	0.002653	-0.00144	0.00579
sigma	0.027	0.0229	0.0301

NORMAL vs CAUCHY



Model	AIC
Normal	-1460
Cauchy	-1363.414

Cauchy distribution could be more suitable for finance data analysis because of the **heavy tails probabilities**, which decay much more slowly.

This would need further analysis in order to make a final decision on the model.

GOALS: ASSIGNMENT 3

Mixture models

1) Fit a **normal mixture model** :
   - 2 components
   - 3 components
2) **Compare models**
3) Report **confidence interval** for the parameters
4) **Profile likelihood** of one of the variance parameters.
5) **Reparametrize** the model to obtain one maximum

HMM models

1) Fit **normal Hidden Markov Model** with 2 and 3 states
2) Find CI 95% for **working parameters** and report **natural parameters** and their CI 95%
3) Plot long term distribution and 1-step ahead distribution - Forecasting
4) Discuss how to do short term prediction

MIXTURE MODELS
Fit a normal mixture model

Unconstrained optimizer : `nlm`

Natural parameters

$$\sigma_i = \exp(\rho_i),\ i = 1,\ldots,m$$

$$\delta_i = \frac{\exp(\tau_i)}{1+\sum_{j=2}^{m}\exp(\tau_j)}\ ,\ \ i = 2,\ldots,m$$

$$\delta_1 = 1 - \sum_{j=2}^{m}\delta_j$$

Working parameters

$$\rho_i = \log(\sigma_i)\ ,\ i = 1,\ldots,m$$

$$\tau_i = \log\left(\frac{\delta_i}{1-\sum_{j=2}^{m}\delta_j}\right),\ i = 2,\ldots,m$$

2 Components :

$$\delta_1 N(\mu_1,\sigma_1^2) + \delta_2 N(\mu_2,\sigma_2^2)$$

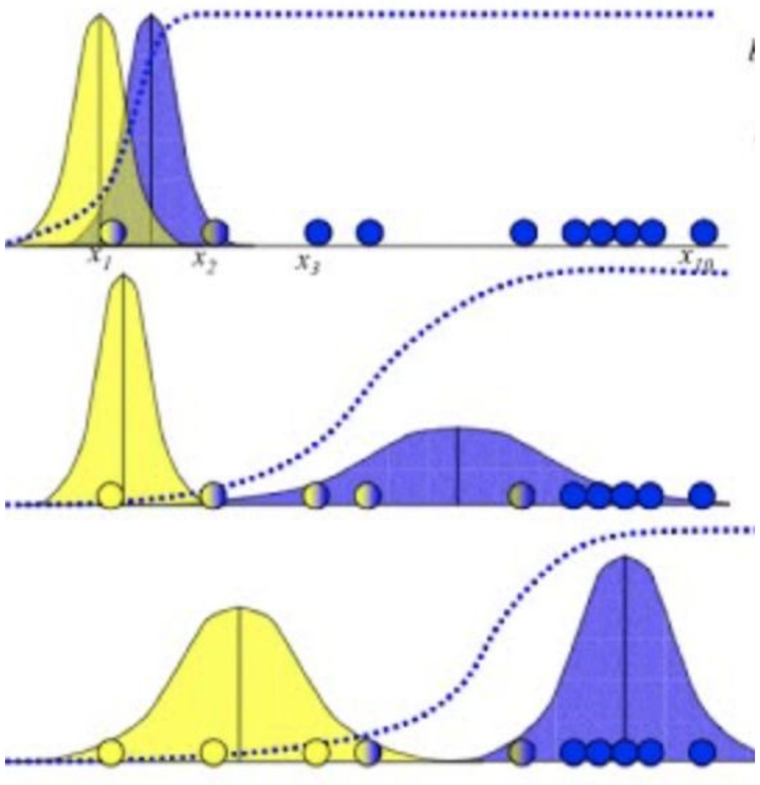
3 Components :

$$\delta_1 N(\mu_1,\sigma_1^2)+\delta_2 N(\mu_2,\sigma_2^2)+\delta_3 N(\mu_3,\sigma_3^2)$$

Likelihood (m components)

$$logL(\theta;y) = \sum_i log \sum_{m=1}^{M} \delta_m N_m(y_i|\mu_m,\sigma_m^2)$$

EM algorithm for mixture models



V. Lavrenko, 2014

MIXTURE MODELS
Fit a normal mixture model

Unconstrained optimizer : `nlm`

Natural parameters

$$\sigma_i = \exp(\rho_i),\ i = 1,\ldots,m$$

$$\delta_i = \frac{\exp(\tau_i)}{1+\sum_{j=2}^{m}\exp(\tau_j)}\ ,\ i = 2,\ldots,m$$

$$\delta_1 = 1 - \sum_{j=2}^{m}\delta_j$$

Working parameters

$$\rho_i = \log(\sigma_i)\ ,\ i = 1,\ldots,m$$

$$\tau_i = \log\left(\frac{\delta_i}{1-\sum_{j=2}^{m}\delta_j}\right)\ ,\ i = 2,\ldots,m$$

Model	m	AIC
Normal	1	-1460
Normal	2	-1489.644
Normal	3	-1484.256

2 Components :

$$\delta_1 N(\mu_1,\sigma_1^2) + \delta_2 N(\mu_2,\sigma_2^2)$$

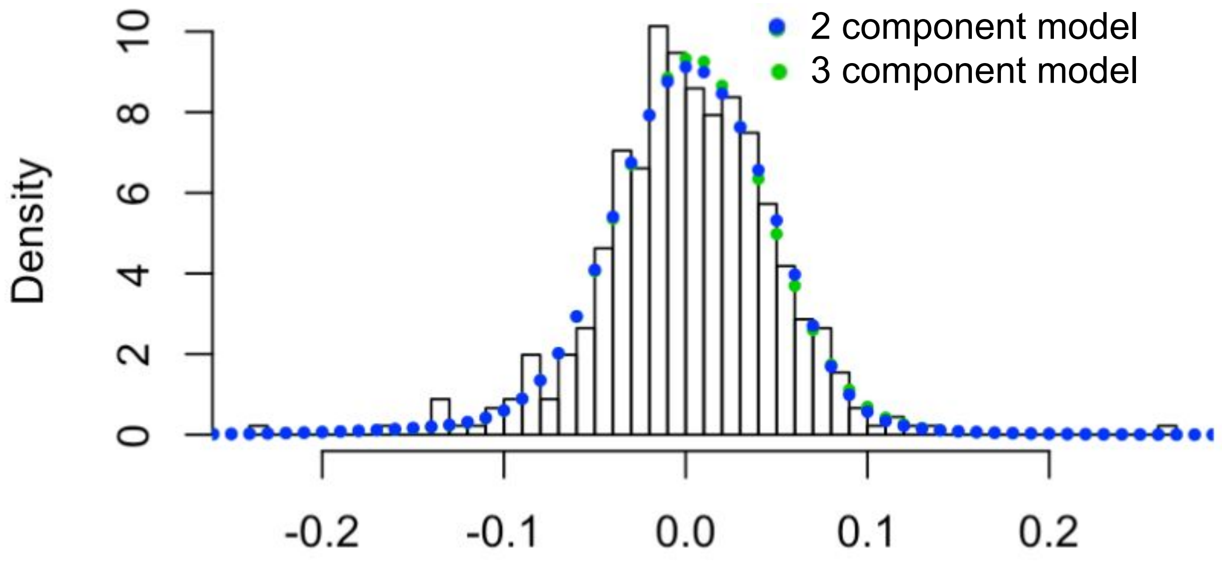
3 Components :

$$\delta_1 N(\mu_1,\sigma_1^2)+\delta_2 N(\mu_2,\sigma_2^2)+\delta_3 N(\mu_3,\sigma_3^2)$$

Likelihood (m components)

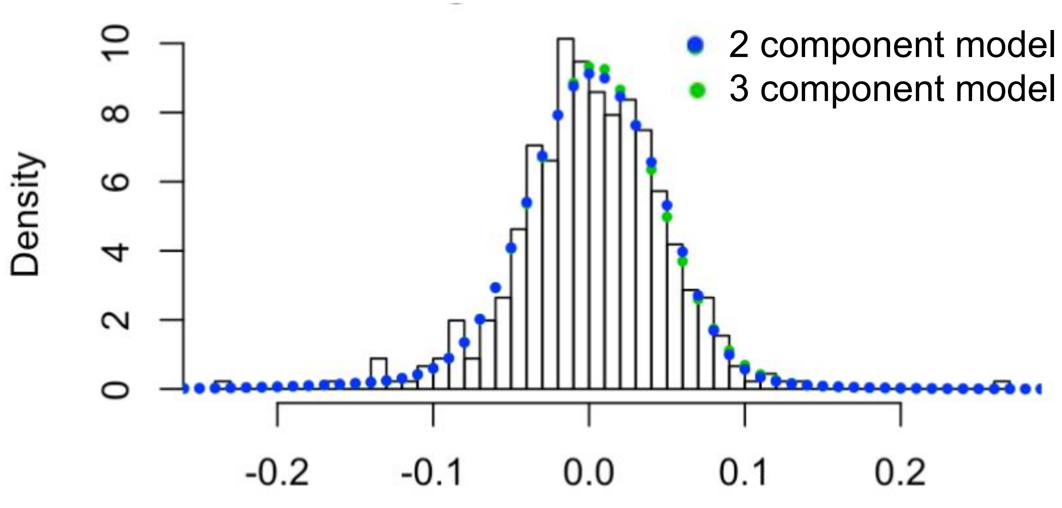
$$logL(\theta; y) = \sum_i log \sum_{m=1}^{M} \delta_m N_m(y_i|\mu_m,\sigma_m^2)$$

Estimated normal distributions



MIXTURE MODELS
Compare models and report CI

Model	m	AIC
Normal	1	-1460
Normal	2	-1489.644
Normal	3	-1484.256



Parameter estimation and CI for m=2

Wald confidence intervals of working parameters:

$$CI(\sigma_i) = \exp\left(\hat{\rho}_i \pm z_{1-\frac{\alpha}{2}} \cdot se\left(\hat{\rho}_i\right)\right), \; i = 1,\ldots,k$$

Wald interval simulation from distribution

$$\hat{\boldsymbol{\theta}} \sim N(\boldsymbol{\theta}, \mathcal{I}^{-1}(\boldsymbol{\theta}))$$

CI from quantiles of 100.000 samples from $N(\hat{\boldsymbol{\theta}}, I^{-1}(\hat{\boldsymbol{\theta}}))$
Transformed back to natural deltas.

	Parameter (N)	Confidence interval (0.025 - 0.975)
μ_1	0.0039	[-0.0007570256 , - 0.0086514630]
μ_2	-0.0251	[-0.06722030 , 0.01692244]
σ_1	0.04046	[0.03592759 , 0.04557654]
σ_2	0.09472	[0.06251529 , 0.14351169]
δ_1	0.9147814	[0.7210504 , 0.9778096]
δ_2	0.08521855	[0.02219040 , 0.27894959]

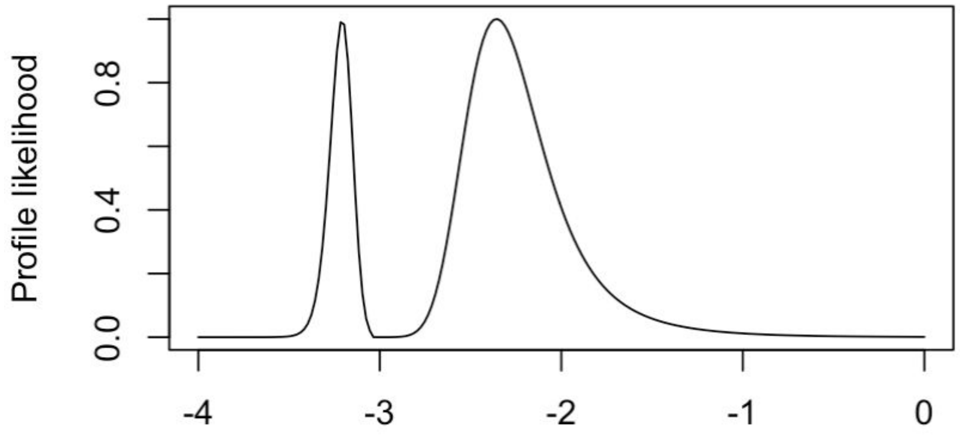
MIXTURE MODELS
Profile Likelihood and reparametrization

Profile Likelihood - Nuissance parameter

$$logL(\hat{\mu}_1, \hat{\mu}_2, \sigma_1^2, \hat{\sigma}_2^2; y) =$$

$$\sum_i log\delta_1 N(y_i|\hat{\mu}_1, \sigma_1^2) + log\delta_2 N(y_i|\hat{\mu}_2, \hat{\sigma}_2^2)$$

	Parameter (W)	Confidence interval (0.025 - 0.975)
σ_1	-3.2073	[-3.326250 , -3.088362]
σ_2	-2.3568	[-2.475785 , -2.237898]



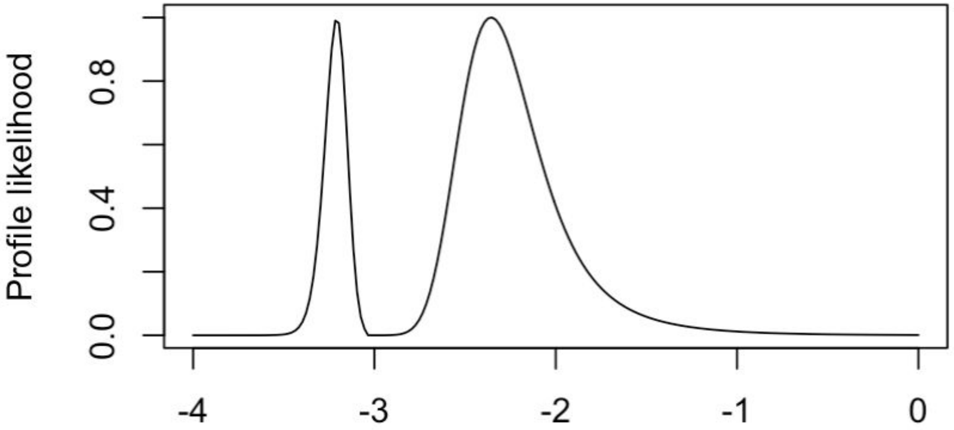
MIXTURE MODELS
Profile Likelihood and reparametrization

Profile Likelihood - Nuissance parameter

$$logL(\hat{\mu}_1, \hat{\mu}_2, \sigma_1^2, \hat{\sigma}_2^2; y) =$$

$$\sum_i log\delta_1 N(y_i|\hat{\mu}_1, \sigma_1^2) + log\delta_2 N(y_i|\hat{\mu}_2, \hat{\sigma}_2^2)$$

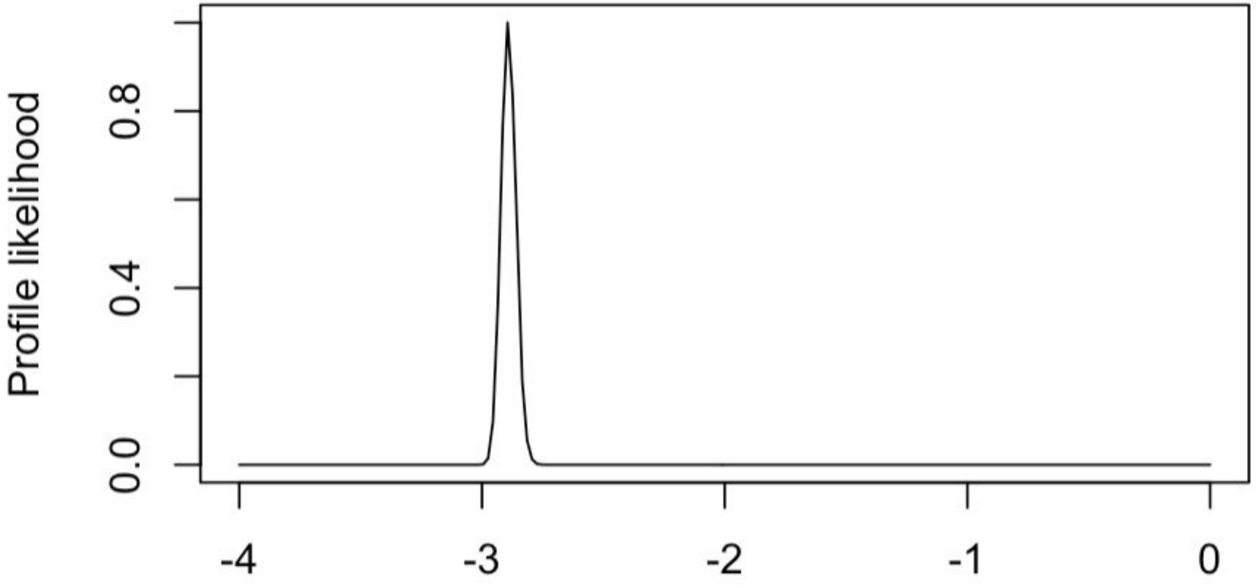
	Parameter (W)	Confidence interval (0.025 - 0.975)
σ_1	-3.2073	[-3.326250 , -3.088362]
σ_2	-2.3568	[-2.475785 , -2.237898]



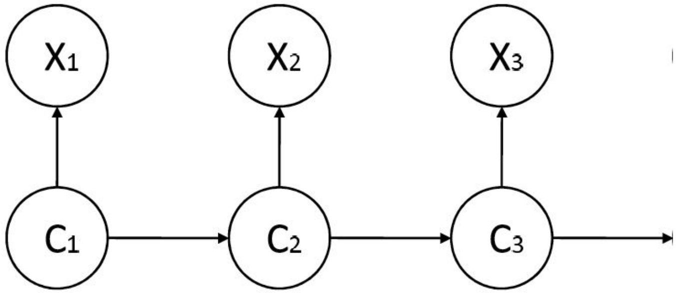
Profile Likelihood - Reparametrization

$$logL(\hat{\mu}_1, \hat{\mu}_2, \sigma_1^2, \boxed{\hat{\sigma}_2^2+\sigma_1^2}; y)$$

$$\sum_i log\delta_1 N(y_i|\hat{\mu}_1, \sigma_1^2) + log\delta_2 N(y_i|\hat{\mu}_2, \boxed{\hat{\sigma}_2^2 + \sigma_1^2})$$



HMM Normal models



Natural to working parameters

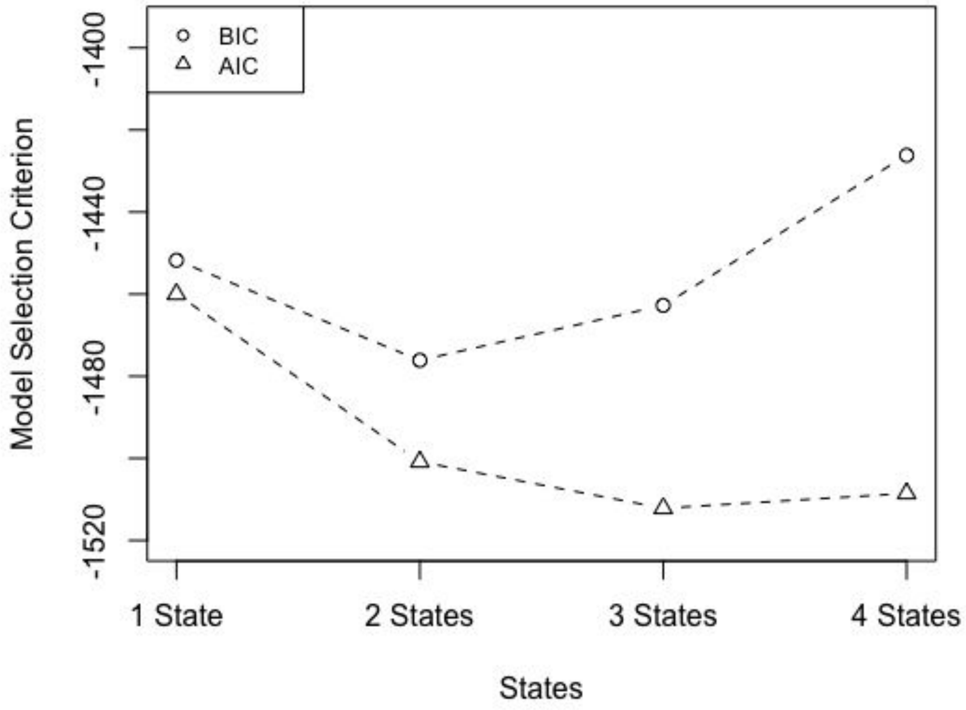
$$\mu_t = \mu$$
$$\sigma_t = \log(\sigma)$$
$$\tau_{ij} = \log\left(\frac{\gamma_{ij}}{1-\sum_{k\neq i}\gamma_{ik}}\right), i = 1, \dots, m \ \ , j = 2, \dots, m$$

Working to natural parameters

$$\mu = \mu_t$$
$$\sigma = \exp(\sigma_t)$$
$$\gamma_{ij} = \frac{\rho_{ij}}{1+\sum_{k\neq i}\exp(\tau_{ik})}, i, j = 1, \dots, m$$

where

$$\rho_{ij} = \begin{cases} \exp(\tau_{ik}) & i \neq j \\ 1 & i = j \end{cases}$$

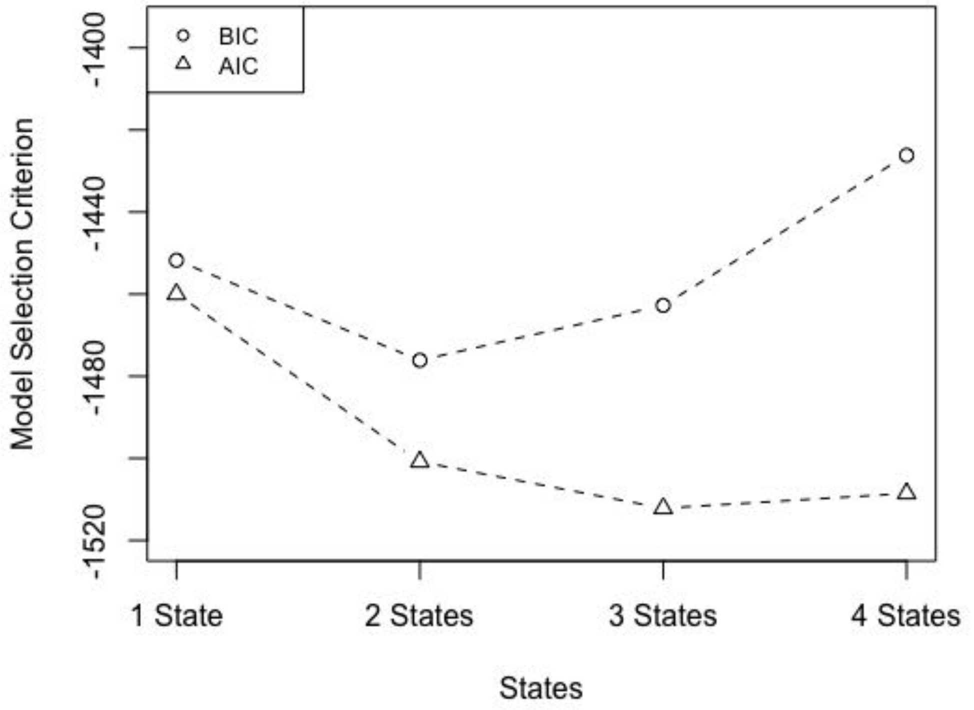


# states	Degrees of freedom	Log-Likelihood	AIC
1	2	731.9998	-1460.000
2	6	756.4172	-1500.834
3	**12**	**768.0791**	**-1512.158**
4	20	774.2719	-1508.544

HMM Normal models

```r
norm.HMM.mllk <- function(parvect,x,m,...)
{
  #    print(parvect)
  if(m==1) return(-sum(dnorm(x, parvect[1], exp(parvect[2]), log=TRUE)))
  n           <- length(x)
  pn          <- norm.HMM.pw2pn(m,parvect)

  allprobs    <- matrix(nrow = n, ncol = m)
  for (j in 1:m){
    allprobs[,j] = dnorm(x, pn$mu[j], pn$sigma2[j])
  }
  allprobs    <- ifelse(!is.na(allprobs),allprobs,1)
  lscale      <- 0
  foo         <- pn$delta
  for (i in 1:n)
  {
    foo    <- foo%*%pn$gamma*allprobs[i,]
    sumfoo <- sum(foo)
    lscale <- lscale+log(sumfoo)
    foo    <- foo/sumfoo
  }
  mllk        <- -lscale
  mllk
}
```



# states	Degrees of freedom	Log-Likelihood	AIC
1	2	731.9998	-1460.000
2	6	756.4172	-1500.834
3	**12**	**768.0791**	**-1512.158**
4	20	774.2719	-1508.544

HMM Model with 3 states

Working parameters

	Estimate	2.5%	97.5%
mu1	0.0119	0.0039	0.01993
mu2	-0.0026	-0.0078	0.0026
mu3	-0.0332	-6.634e-02	-5.89e-05
sigma21	-3.1104	-3.268	-2.9528
sigma22	-3.5034	-3.6368	-3.37
sigma23	-2.4825	-2.7552	-2.21
tau21	-28.3362	NaN	NaN
tau31	-1.018	-2.1914	0.1555
tau12	-4.0342	-5.3756	-2.6927
tau32	-19.4691	-21.3123	-17.626
tau13	-3.0867	-4.908	-1.2654
tau23	-3.8643	-5.271	-2.4576

Natural parameters

$\mu_1 = 0.0119$ $\mu_2 = -0.0026$ $\mu_3 = -0.0332$

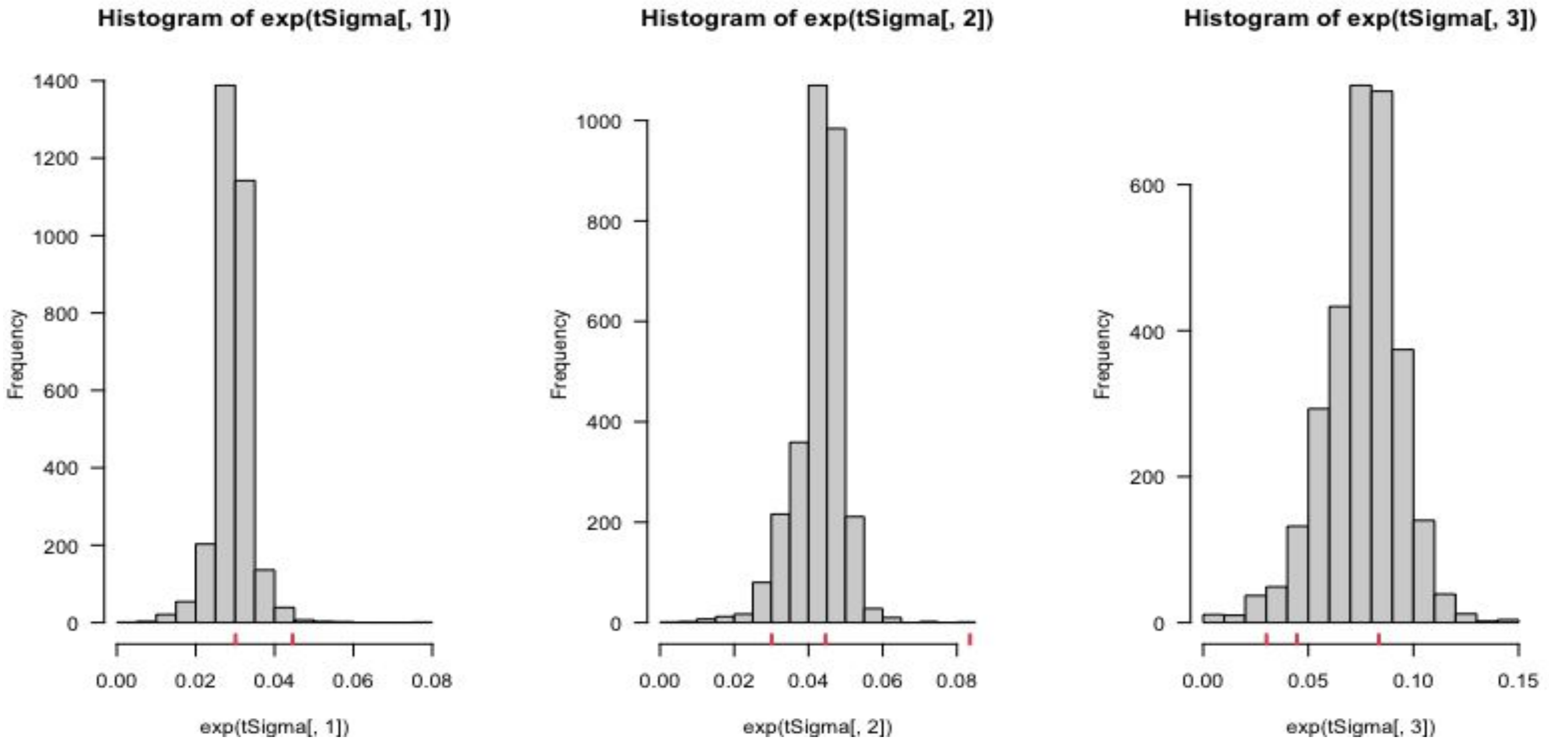
$\sigma_1 = 0.0446$ $\sigma_2 = 0.03$ $\sigma_3 = 0.0835$

$\delta = [\ 0.4915 \quad 0.3982 \quad 0.1103\ ]$

$$\tau_1 = \begin{matrix} 0.9404 & 0.0166 & 0.0429 \\ 0.0000 & 0.9795 & 0.0205 \\ 0.2654 & 0.0000 & 0.7346 \end{matrix}$$

HMM Model with 3 states

NATURAL PARAMETERS CI 95% BOOTSTRAP WITH K= 3000



	MLE	2.5%	97.5%
σ_1	0.04458	0.0198	0.0389
σ_2	0.03	0.0291	0.0534
σ_3	0.08353	0.034	0.1082

	MLE	2.5%	97.5%
μ_1	0.01193	-0.0110	0.0119
μ_2	-0.00258	-0.0048	0.0311
μ_3	-0.03319	-0.1495	0.0088

STEPS:

1. Generate a sample from the MLE
2. Fit new model to the sample
3. Store the MLE of the parameters estimated in the new distribution

Repeat 1-3 k times

Make short term predictions

- Using **Viterbi algorithm** to decode state sequence until now, and predict next week's state
- Look at previous observations which were observed after the **same state transition** as the upcoming one

References

Pawitan Y. In All Likelihood: Statistical Modelling and Inference Using Likelihood. OUP Oxford; 2001. (Oxford science publications)

Code for the project can be found at Statistical Modelling

DTU

Long term and 1-step ahead model forecast

```
normal.HMM.forecast <- function(xf, h=1, m, x, mod){
  n <- length (x)
  nxf <- length (xf)
  dxf <- matrix (0, nrow =h, ncol = nxf)
  foo <- mod$delta * dnorm (x[1] , mod$mu, mod$sigma2 )
  sumfoo <- sum (foo)
  lscale <- log ( sumfoo )
  foo <- foo / sumfoo
  for (i in 2:n){
    foo <- foo %*% mod$gamma * dnorm(x[i], mod$mu, mod$sigma2)
    sumfoo <- sum( foo)
    lscale <- lscale + log ( sumfoo )
    foo<- foo / sumfoo
  }
 for (i in 1:h)
 {
    foo <- foo %*% mod$gamma
    for (j in 1: m ) dxf[i ,] <- dxf[i ,] +
      foo [j]* dnorm (xf , mod$mu[j], mod$sigma2)
    }
  return ( dxf)
}
```

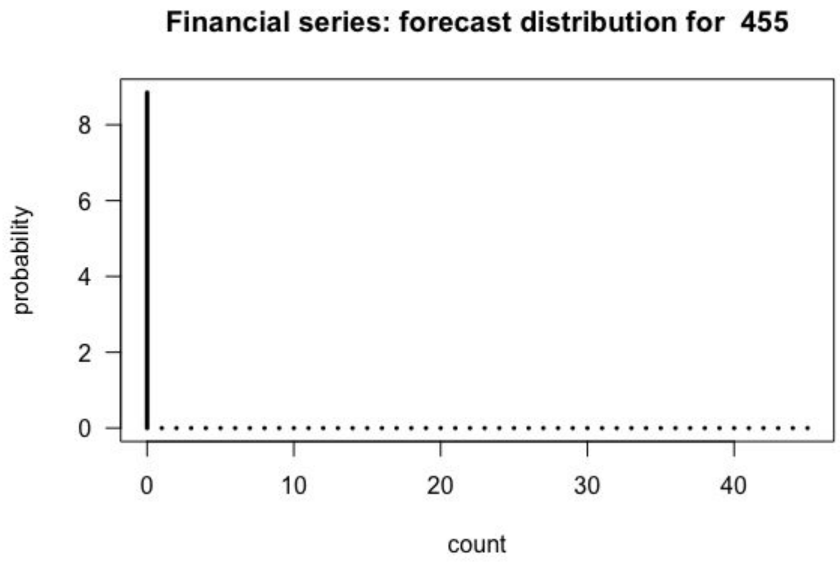
We wanted to do 1-step ahead with the function on the left, to compute the marginal distribution of the data.

To do 1-step ahead we would set the h=1, 1 year ahead and we would use our 3-states model (mod3s) with the argument stationary = TRUE.

On the other hand to do the the long term prediction, we use our initial 3-state model, this time not taking the stationary argument (fit3). We use our mu, and sigma2 and extract the marginal distribution with the code below:

```
m < -3
xf <- 0:45
mu <- fit3$mu
sigma2 <- fit3$sigma2
delta <- solve (t( diag (m)- fit3$gamma +1) ,rep (1,m))
dstat <- numeric ( length (xf))
for (j in 1:m) dstat <- dstat + delta [j]* dnorm(xf , mu[j], sigma2[j])
```

Long term and 1-step ahead model forecast



1 step ahead

Long term prediction