

# Deep Survival Analysis by Ranganath

*Paidamoyo Chapfuwa*

*1/30/2017*

## Deep Survival Analysis : EHR

- Hierarchical Generative Approach
  - survival times and covariates modeled jointly conditional on latent processes
  - records align by failure time rather than entry time
  - Each observation has L layers of latent variables
- (t,c) pairs of positive times and binary censoring status
- Limitations to traditional approaches( Cox(<http://www.scharp.org/users/yqchen/class/Cox1972JRSSB.pdf>) & Kaplan Meir(<http://www.biecek.pl/statystykaMedyczna/2281868.pdf>))
  - EHR highdimensional and sparse (missing covariates)
  - Traditional methods require aligning all patients on a synchronizing event(i.e entry to clinical trial)
  - Regression based approaches assume linear function
- Generative Model:  $\mathbf{x}$  set of covariates,  $\beta$  data parameters with prior  $p(\beta)$ ,  $k = 2$  fixed scalar,  $n$  index of observation
  - $b \sim \text{Normal}(0, \sigma_b)$  (variance =1)
  - $a \sim \text{Normal}(0, \sigma_w)$  (variance =1)
  - $z_n \sim \text{DEF}(\mathbf{W})$  (Gaussian)
  - $x_n \sim p(\cdot | \beta, z_n)$  (Real values are student-t distribute, count valued are bernoulli)
  - $t_n \sim \text{Weibull}(\log(1 + \exp(z_n^T a + b)), k)$
  - $p(t|x) = \int_z p(t|z)p(z|x)$
  - $p(z|x) = \frac{p(x|\beta, z)p(\beta)p(z)}{p(x)}$
- Variational Inference Implemented to estimate posterior distribution:
  - Kingma & Welling (2014): Auto-Encoding Variational Bayes (<https://arxiv.org/pdf/1312.6114v10.pdf>)
  - Hoffman (2013): Stochastic Variational Inference (<http://jmlr.org/papers/volume14/hoffman13a/hoffman13a.pdf>)
  - Jordan (1999): An Introduction to Variational Methods for Graphical Models (<https://people.eecs.berkeley.edu/~jordan/papers/variational-intro.pdf>)
  - Ranganath(2014): Black Box Variational Inference (<http://www.cs.columbia.edu/~blei/papers/RanganathGerrishBlei2014.pdf>)
  - Rezende (2014): Stochastic Backpropagation and Approximate Inference in Deep Generative Models (<http://jmlr.csail.mit.edu/proceedings/papers/v32/rezende14-suppl.pdf>)

## Weibull Distribution

```
#lambda scale
# k shape
# mean =
# t time to failure
lambda <- 10
t <- seq(0, 20, by=0.1)

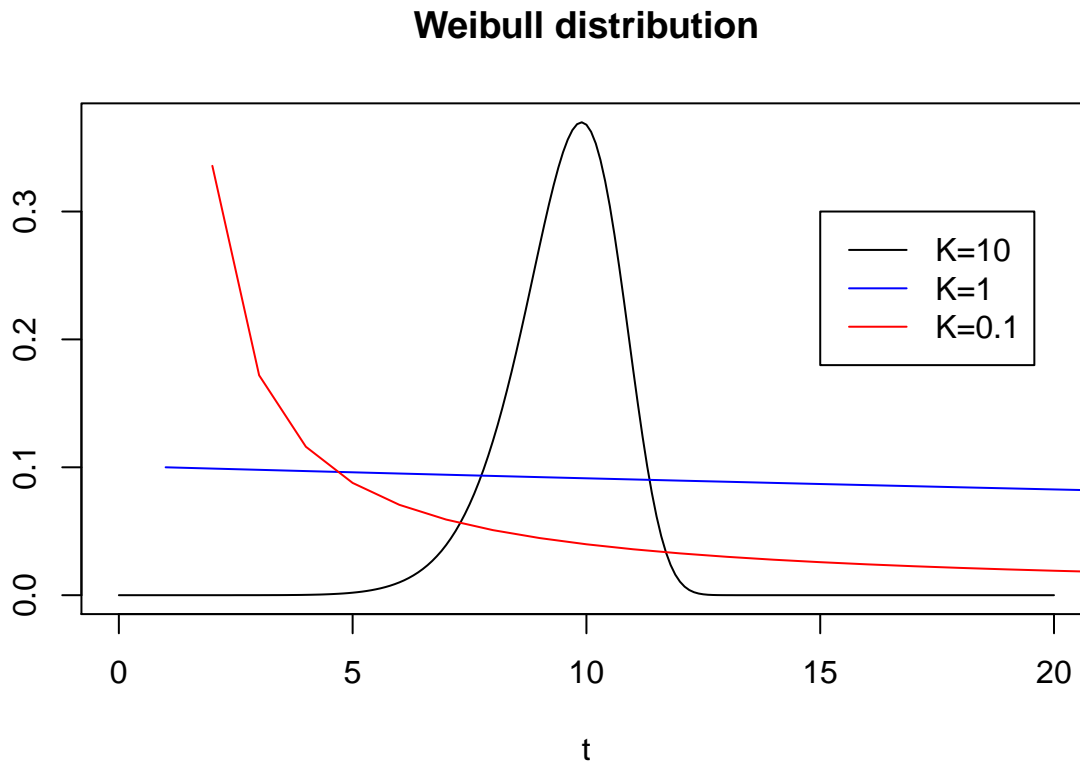
weibul <- function(k) {
  return(k/ lambda * (t/ lambda) ^ (k-1) * exp(-(t/lambda) ^k))
}
```

```

}

plot(t, weibul(10), main = 'Weibull distribution', xlab='t', ylab = '', type='l')
lines(weibul(1), col='blue')
lines(weibul(0.1), col='red')
legend(15, 0.3, legend=c("K=10", "K=1", "K=0.1"), lty = c(1,1,1), col = c('black', 'blue', 'red'))

```



## Student-t distribution (describes samples from a normal distribution)

Arises when estimating mean of a normal distributed population where sample size is small and population standard deviation is unknown, larger sample size resembles a normal distribution.

n observations from a normal distribution results in a t-distribution with  $v = n-1$  degree of freedom, has heavier tails more prone to producing values away from the true mean

```

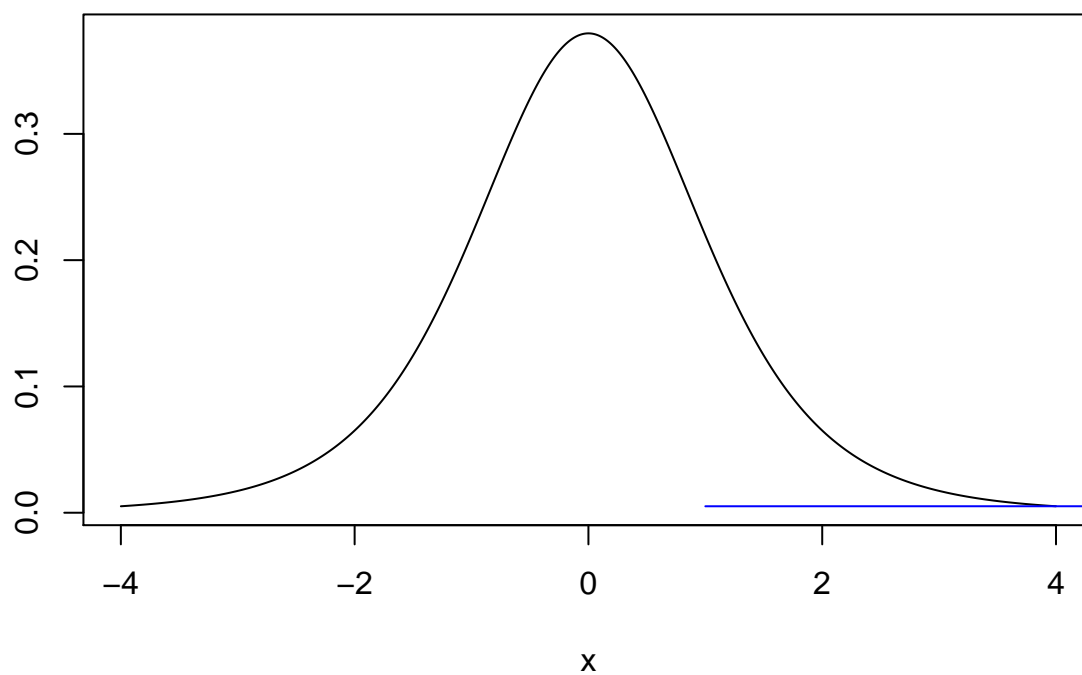
# v = n-1 degree of freedom

x <- seq(-4, 4, by=0.001)
t_dist <- function(v) {
  norm_constant <- gamma((v+ 1)/2) / (sqrt(v * pi) * gamma(v/2))
  return (norm_constant * (1 + x^2/v) ^(-(v + 1)/2 ))
}

plot(x, t_dist(5), main = 'Student-t distribution', xlab='x', ylab = '', type='l')
lines(t_dist(5), col='blue')

```

## Student-t distribution



```
# lines(t_dist(100), col='red')  
# legend(-3, 0.3, legend=c("v=1", "v=5", "v=100"), lty = c(1,1,1), col = c('black', 'blue', 'red') )
```