Distribution and Parameter Inference

Il-Chul Moon
Dept. of Industrial and Systems Engineering
KAIST

icmoon@kaist.ac.kr

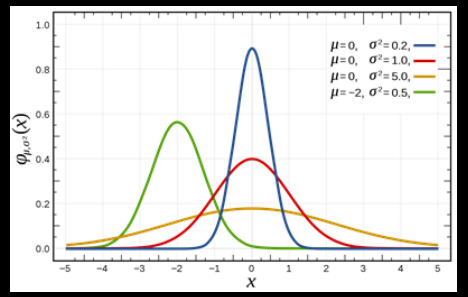
Detour) Dataset Introduction: Credit Score

- This research aimed at the case of customers' default payments in Taiwan
- Dependent variable :
  - a binary variable, default payment (Yes = 1, No = 0), as the response variable.
- Independent variable:
  - X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
  - X2: Gender (1 = male; 2 = female).
  - X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
  - X4: Marital status (1 = married; 2 = single; 3 = others).
  - X5: Age (year).
  - X6 - X11: History of past payment.
    - We tracked the past monthly payment records (from April to September, 2005) as follows: X6 = the repayment status in September, 2005; X7 = the repayment status in August, 2005; . . .;X11 = the repayment status in April, 2005.
    - The measurement scale for the repayment status is: -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.
  - X12-X17: Amount of bill statement (NT dollar).
    - X12 = amount of bill statement in September, 2005; X13 = amount of bill statement in August, 2005; . . .; X17 = amount of bill statement in April, 2005.
  - X18-X23: Amount of previous payment (NT dollar)
    - X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .;X23 = amount paid in April, 2005.
- Yeh, I. C., & Lien, C. H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Systems with Applications, 36(2), 2473-2480.
- Dua, D. and Graff, C. (2019). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

Detour) Bin Size Selection

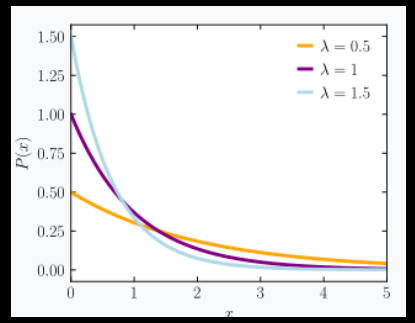
© wikipedia

Normal Dist

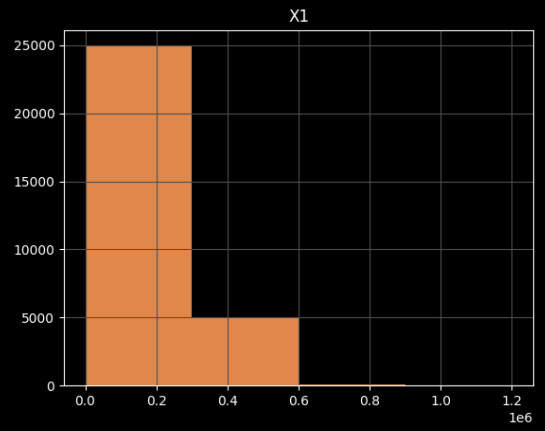


- Which bin size would be better?
  - How to setup the bin size?
  - Noise vs. Complexity
- Some systematic selection by formula
  - Square-root choice : $k = \lceil \sqrt{n} \rceil$
  - Sturge's formula : $k = \sqrt{\log_2 n} + 1$
  - There are many more formulas….
- What's the ultimate goal of drawing histogram?
  - Identifying the data distribution

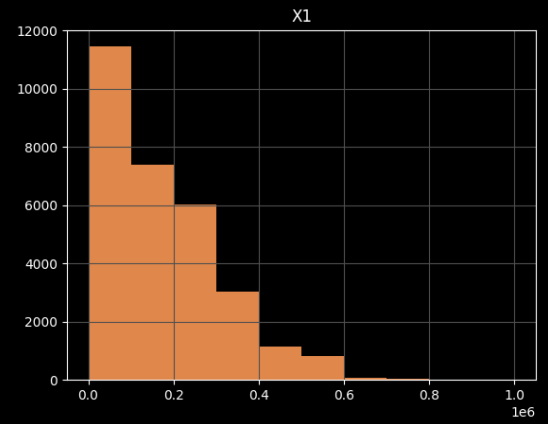
Exp. Dist



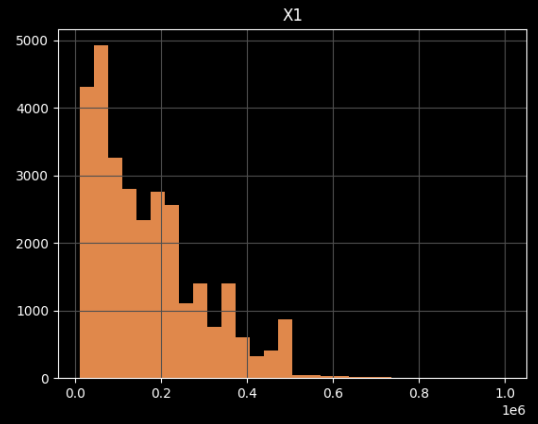
4 Bins



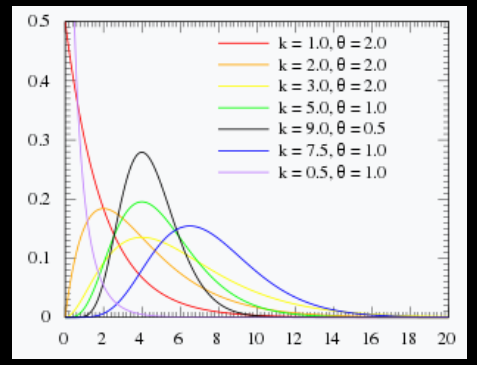
11 Bins



30 Bins



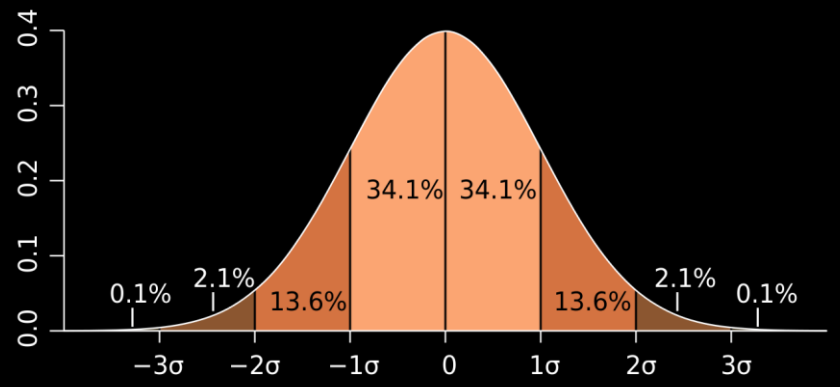
Gamma Dist

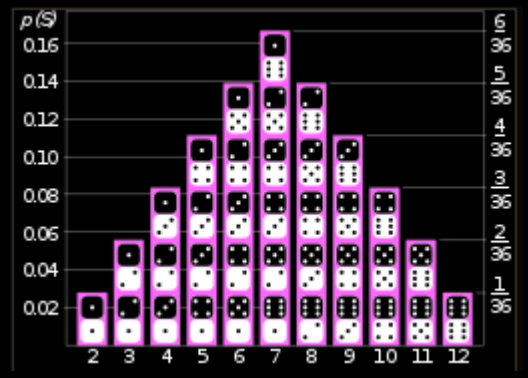


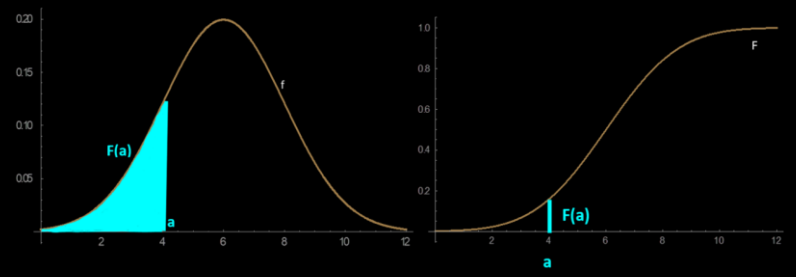
(c)wikipedia

Distribution

- (Probability) Distribution
  - Mathematical description of the probabilities of events
    - Events are the subsets of the sample space, Ω.
  - Discrete random variable
    - Probability mass function (PMF)
  - Continuous random variable
    - Probability density function (PDF)
- Some terminologies
  - Random variable
  - Probability measure
  - Cumulative distribution function
  - Support
  - Parameters

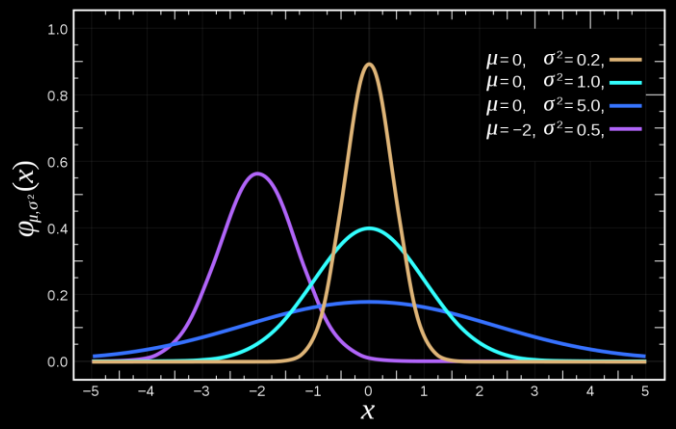






(c)wikipedia

(Univariate) Normal Distribution

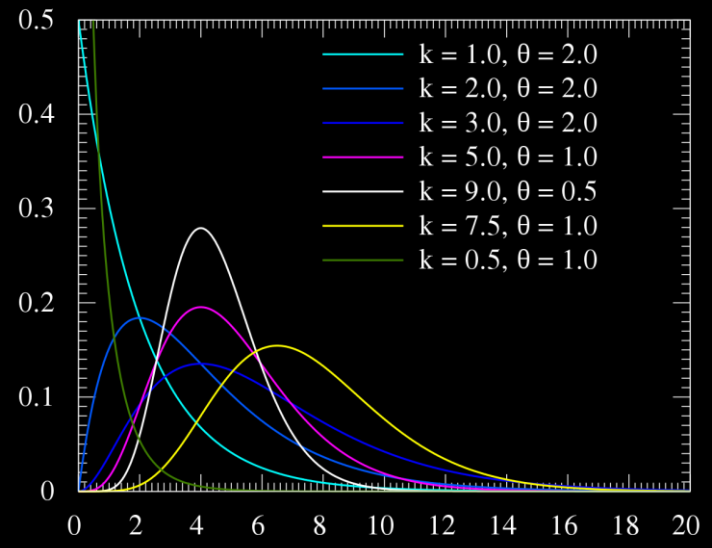


- Application
  - Double-sided long tailed distribution of continuous values
  - Many applications of natural and social sciences
- Key characteristics
  - Probability density function : $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$
  - Notation : $X \sim N(\mu, \sigma^2)$
  - Parameters : $\mu \in R$, $\sigma^2 \in R_{>0}$
  - Support : $x \in R$
  - Mean : μ
  - Variance : σ^2
- Key variations
  - Standard normal distribution : $\mu = 0$, $\sigma = 1$
  - Alternative parameterization: precision $\tau = \frac{1}{\sigma}$ can be used in substitution of σ

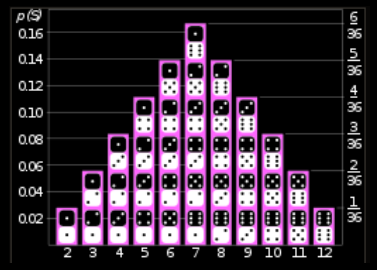
(c)wikipedia

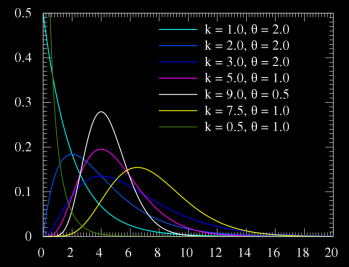
Gamma Distribution

- Application
  - Some econometric applications
  - Waiting time modeling
  - Life time testing
- Key characteristics
  - Probability density function : $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$
    - $\Gamma(k)$: Gamma function
      - In discrete version : $\Gamma(k) = (k-1)!$
  - Notation : $X \sim \Gamma(\alpha, \beta)$
  - Parameters : $\alpha \in R_{>0}, \beta \in R_{>0}$
    - α : shape parameter, β: inverse scale parameter
  - Support : $x \in R_{>0}$
  - Mean : $\frac{\alpha}{\beta}$
  - Variance : $\frac{\alpha}{\beta^2}$



$$\alpha = k, \beta = \frac{1}{\theta}$$

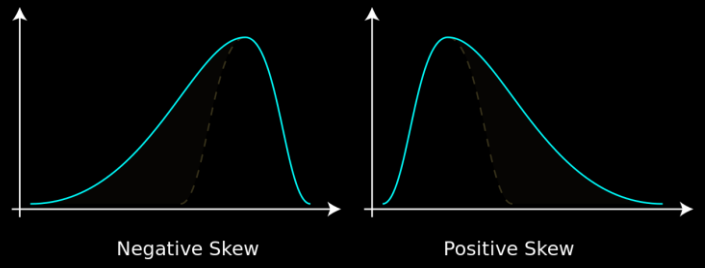


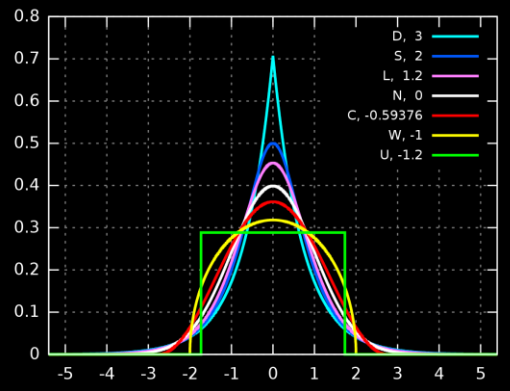


(c)wikipedia

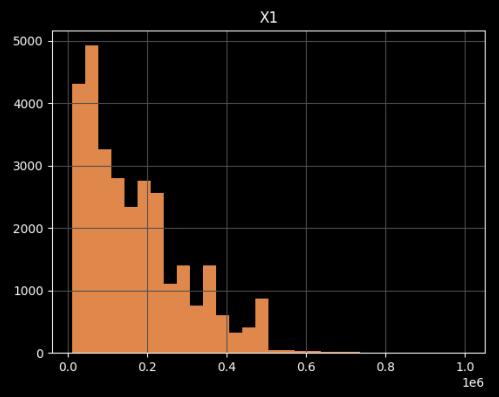
Moment Statistics

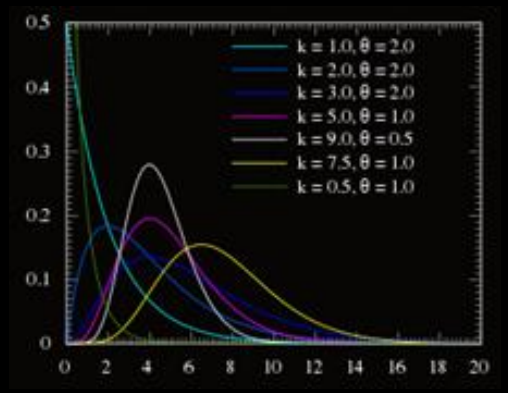
- Summary statistics
  - A summary of distribution providing the shape, the scale, the skewness, etc
- (Raw) moments
  - Discrete distribution: $E[X^k] = \sum_{x \in S} x^k p(x)$
  - Continuous distribution: $E[X^k] = \int_{-\infty}^{\infty} x^k f(x) dx$
- k-th central moment
  - $\mu_k = E[(X - \mu)^k]$
- Each moment is related to semantic meanings
  - First moment: mean $= E[X]$
  - Second moment: variance $= E[(X - \mu)^2]$
  - Third moment: skewness $= \frac{E[(X-\mu)^3]}{\sigma^3}$
  - Fourth moment: kurtosis $= \frac{E[(X-\mu)^4]}{\sigma^4}$



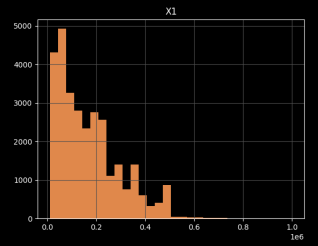


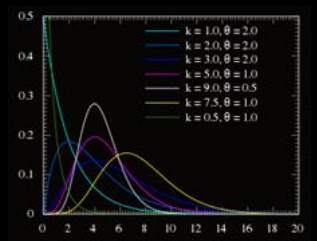
How to Fit Distribution to Data?





- Method of moments
  - Matching between the data and the parameterized distribution
    - Update the parameter, so the distribution shape will be similar to the data histogram
  - Let's fit the distribution by matching the moment statistics
    - Moments describe the summary of the distribution shape
    - Therefore, matching moments will result in the same distribution shape
- Method of maximum likelihood





Method of Moments

- Matching the sample moments to the distribution moments to derive k unknown parameters, $\theta_1 \dots \theta_k$ of probability density function
  - Distribution moments
    - $\mu_1 \equiv E[X] = g_1(\theta_1, \dots, \theta_k)$
    - $\dots$
    - $\mu_k \equiv E[X^k] = g_k(\theta_1, \dots, \theta_k)$
      - g functions are derived from f to define k-th moment
        - Is there any common form of g? → Moment generating function (MGF)
        - Moments define the shape of distribution → MGF can be an alternative specification for probability distribution
  - Sample moments
    - $\hat{\mu}_j = \frac{1}{n}\sum_{i=1}^{n} x_i^j$
- Moment matching between the distribution moments and the sample moments

Key characteristics

Probability density function : $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$

Notation : $X \sim N(\mu, \sigma^2)$
Parameters : $\mu \in R$, $\sigma^2 \in R_{>0}$
Support : $x \in R$
Mean : μ
Variance : σ^2

Method of Moments for Gaussian Dist.

- Moment matching : $\mu_j \equiv E\left[X^j\right] = g_j(\theta_1, \dots, \theta_k) \rightarrow \hat{\mu}_j = \frac{1}{n}\sum_{i=1}^{n} x_i^j$
- Moment matching for Gaussian distribution
  - 1st moment
    - $\mu_1 \equiv E[X^1] = E[X] = g_1(\mu, \sigma^2) = \mu \rightarrow \hat{\mu}_1 = \frac{1}{n}\sum_{i=1}^{n} x_i^1 = \frac{1}{n}\sum_{i=1}^{n} x_i$
    - $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i$
  - 2nd moment
    - $\mu_2 \equiv E[X^2] = g_2(\mu, \sigma^2) = var(X) + \{E[X]\}^2 = \sigma^2 + \mu^2 \rightarrow \hat{\mu}_2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2$
      - Do you rememeber $var(X) = E[X^2] - \{E[X]\}^2$
    - $\hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \mu^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \left\{\frac{1}{n}\sum_{i=1}^{n} x_i\right\}^2 = \frac{1}{n}\sum_{i=1}^{n}[x_i - \hat{\mu}]^2$
- Method of moments results in the estimated parameter from data, X.

Key characteristics

Probability density function : $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$

$\Gamma(k)$: Gamma function

In discrete version : $\Gamma(k) = (k-1)!$

Notation : $X \sim \Gamma(\alpha, \beta)$

Parameters : $\alpha \in R_{>0}, \beta \in R_{>0}$

α : shape parameter, β: inverse scale parameter

Support : $x \in R_{>0}$

Mean : $\frac{\alpha}{\beta}$

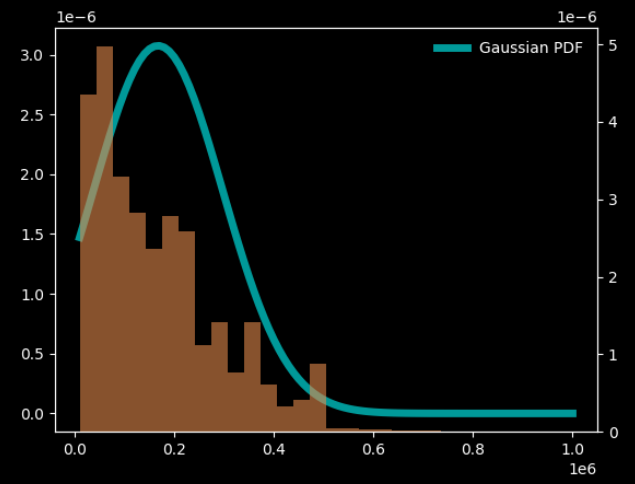
Variance : $\frac{\alpha}{\beta^2}$

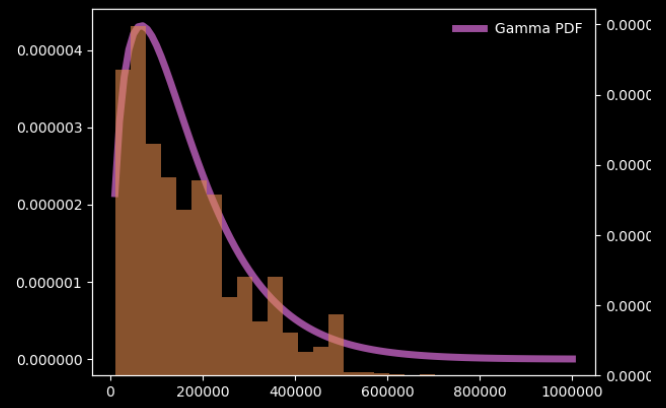
Method of Moments for Gamma Dist.

- Moment matching : $\mu_j \equiv E\left[X^j\right] = g_j(\theta_1, \dots, \theta_k) \rightarrow \hat{\mu}_j = \frac{1}{n}\sum_{i=1}^n x_i^j$
- Moment matching for gamma distribution
  - 1st moment
    - $\mu_1 \equiv E[X^1] = E[X] = g_1(\alpha, \beta) = \frac{\alpha}{\beta} \rightarrow \hat{\mu}_1 = \frac{1}{n}\sum_{i=1}^n x_i^1 = \frac{1}{n}\sum_{i=1}^n x_i$
    - $\frac{\hat{\alpha}}{\hat{\beta}} = \frac{1}{n}\sum_{i=1}^n x_i = \bar{x}$
  - 2nd moment
    - $\mu_2 \equiv E[X^2] = g_2(\alpha, \beta) = var(X) + \{E[X]\}^2 = \frac{\alpha}{\beta^2} + \frac{\alpha^2}{\beta^2} \rightarrow \hat{\mu}_2 = \frac{1}{n}\sum_{i=1}^n x_i^2$
      - Do you rememeber $var(X) = E[X^2] - \{E[X]\}^2$
    - $\frac{\hat{\alpha}}{\hat{\beta}^2} + \frac{\hat{\alpha}^2}{\hat{\beta}^2} = \frac{1}{n}\sum_{i=1}^n x_i^2$
  - $\bar{x}\frac{1}{\hat{\beta}} + \bar{x}^2 = \frac{1}{n}\sum_{i=1}^n x_i^2 \rightarrow \frac{1}{\hat{\beta}} = \frac{1}{n\bar{x}}\sum_{i=1}^n x_i^2 - \bar{x} \rightarrow \hat{\beta} = \frac{n\bar{x}}{\sum_{i=1}^n x_i^2 - n\bar{x}^2}$
  - $\hat{\alpha} = \frac{n\bar{x}^2}{\sum_{i=1}^n x_i^2 - n\bar{x}^2} = \frac{\hat{\mu}_1^2}{\hat{\mu}_2 - \hat{\mu}_1^2}, \hat{\beta} = \frac{n\bar{x}}{\sum_{i=1}^n x_i^2 - n\bar{x}^2} = \frac{\hat{\mu}_1}{\hat{\mu}_2 - \hat{\mu}_1^2}$
- Method of moments results in the estimated parameter from data, X.

Result from Method of Moments

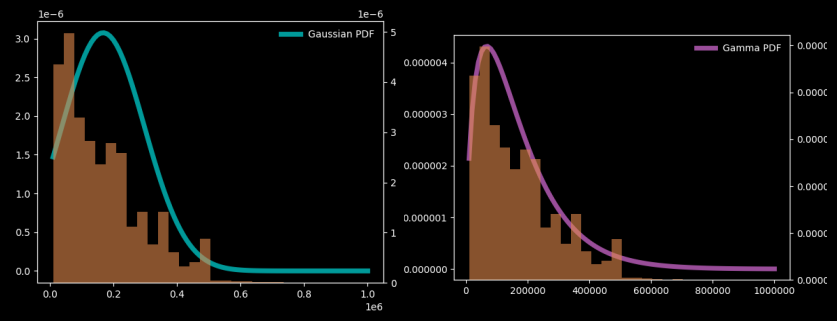
- Fitting results from method of moments
- Which distribution provides a better fitting?
  - Gaussian distribution
  - Gamma distribution
- Both could be problematic in over- or under-estimation on density for some support period
- Calculated results
  - Gaussian distribution
    - Mean: 167489.2389746325
    - Variance: 16833730579.636848
  - Gamma distribution
    - Mean: 167489.2389746325
    - Variance: 16833730579.636848





Implementation of Method of Moments

Added library : scipy 1.9.3



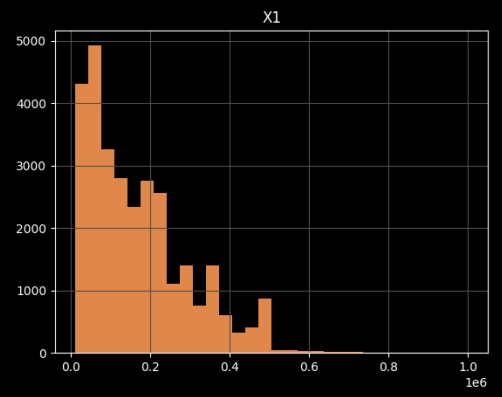

```
from scipy.stats import norm, gamma
```

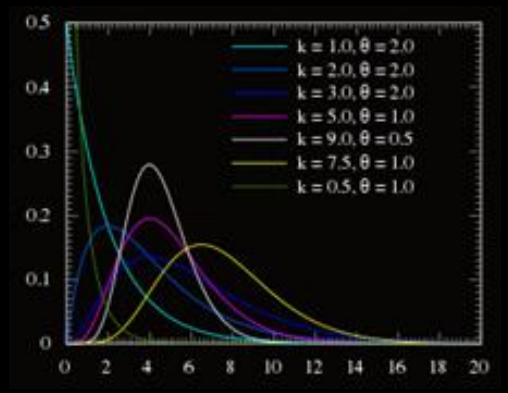
$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} x_i,\ \hat{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n} x_i^2 - \left\{\frac{1}{n}\sum_{i=1}^{n} x_i\right\}^2$$

$$\hat{\alpha} = \frac{\hat{\mu}_1^2}{\hat{\mu}_2 - \hat{\mu}_1^2},\ \hat{\beta} = \frac{\hat{\mu}_1}{\hat{\mu}_2 - \hat{\mu}_1^2}$$

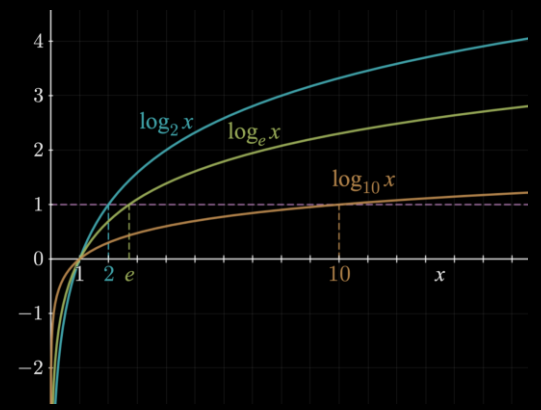
```
dblSampleMoment1 = 0
dblSampleMoment2 = 0
for i in range(len(objData['X1'])):
    dblSampleMoment1 = dblSampleMoment1 + objData['X1'][i]
    dblSampleMoment2 = dblSampleMoment2 + math.pow(objData['X1'][i],2)
dblSampleMoment1 = dblSampleMoment1 / float(len(objData['X1']))
dblSampleMoment2 = dblSampleMoment2 / float(len(objData['X1']))

dblMu = dblSampleMoment1
dblSigmaSquared = dblSampleMoment2 - math.pow(dblSampleMoment1,2)
dblAlpha = math.pow(dblSampleMoment1,2) / (dblSampleMoment2 - math.pow(dblSampleMoment1,2))
dblBeta = dblSampleMoment1 / (dblSampleMoment2 - math.pow(dblSampleMoment1,2))

fig, ax1 = plt.subplots(1, 1)
ax2 = ax1.twinx()

x = np.linspace(min(objData['X1']), max(objData['X1']), 100)
ax1.plot(x, gamma.pdf(x,a=dblAlpha,scale=1/dblBeta), 'g-', lw=5, alpha=0.6, label='Gamma PDF')
ax2.hist(objData['X1'], bins=30, density=True, alpha=0.6)

ax1.legend(loc='best', frameon=False)
plt.show()

fig, ax1 = plt.subplots(1, 1)
ax2 = ax1.twinx()

x = np.linspace(min(objData['X1']), max(objData['X1']), 100)
ax1.plot(x, norm.pdf(x,dblMu,math.sqrt(dblSigmaSquared)), 'r-', lw=5, alpha=0.6, label='Gaussian PDF')
ax2.hist(objData['X1'], bins=30, density=True, alpha=0.6)

ax1.legend(loc='best', frameon=False)
plt.show()
```

Maximum Likelihood Estimation





- Method of moments
- Method of maximum likelihood
  - Apply the observed data to the probability distribution function
  - Optimize the parameters to maximize the likelihood
    - Optimization often requires the gradient method
    - Hence, the distribution ideally needs to be converted to the monotonic convex/concave function



MLE of Gaussian Distribution

Probability distribution function : $f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$

Convert PDF to be monotonic concave function : log-likelihood

- Maximize the log-likelihood → Maximize the likelihood
- How to maximize? → Gradient method

- $\theta_{MLE} = \arg\max_{\theta} P(X|\theta) = \arg\max_{\mu,\sigma} \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2} = \arg\max_{\mu,\sigma} \log\left[\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2}\right]$

$$= \arg\max_{\mu,\sigma} \sum_{i=1}^{n} \left\{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2 - \log\sigma\sqrt{2\pi}\right\} = \arg\max_{\mu,\sigma} \sum_{i=1}^{n} \left\{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2 - \log\sigma\right\}$$

- $\frac{\partial}{\partial\mu}\left[\sum_{i=1}^{n}\left\{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2 - \log\sigma\right\}\right] = \sum_{i=1}^{n}\left\{-\frac{1}{2}\times 2\times\left(\frac{x_i-\mu}{\sigma}\right)\times -\frac{1}{\sigma}\right\} = \sum_{i=1}^{n}\left\{\left(\frac{x_i-\mu}{\sigma^2}\right)\right\} = 0$
  - $\rightarrow \sum_{i=1}^{n}\{(x_i-\mu)\} = 0 \rightarrow \sum_{i=1}^{n} x_i - n\mu = 0 \rightarrow \mu = \frac{1}{n}\sum_{i=1}^{n} x_i$
- $\frac{\partial}{\partial\sigma^2}\left[\sum_{i=1}^{n}\left\{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2 - \frac{1}{2}\log\sigma^2\right\}\right] = \sum_{i=1}^{n}\left\{-\frac{1}{2}\times -1\times\frac{(x_i-\mu)^2}{(\sigma^2)^2} - \frac{1}{2\sigma^2}\right\} = \sum_{i=1}^{n}\frac{1}{2}\frac{(x_i-\mu)^2}{(\sigma^2)^2} - \frac{n}{2\sigma^2} = 0$
  - $\rightarrow \sum_{i=1}^{n}(x_i-\mu)^2 - n\sigma^2 = 0 \rightarrow n\sigma^2 = \sum_{i=1}^{n}(x_i-\mu)^2 \rightarrow \sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i-\mu)^2$
- Same result to the method of moments!

MLE of Gamma Distribution

- Probability distribution function : $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$
- $\theta_{MLE} = \arg\max_{\theta} P(X|\theta) = \arg\max_{\mu,\sigma} \prod_{i=1}^{n} \frac{\beta^\alpha}{\Gamma(\alpha)} x_i^{\alpha-1} e^{-\beta x_i}$

$$= \arg\max_{\mu,\sigma} \log\left[\prod_{i=1}^{n} \frac{\beta^\alpha}{\Gamma(\alpha)} x_i^{\alpha-1} e^{-\beta x_i}\right]$$

$$= \arg\max_{\mu,\sigma} \sum_{i=1}^{n} [(\alpha-1)x_i - \log\Gamma(\alpha) + \alpha\log\beta - \beta x_i]$$

- $\frac{\partial}{\partial\beta} \sum_{i=1}^{n} [(\alpha-1)x_i - \log\Gamma(\alpha) + \alpha\log\beta - \beta x_i] = \frac{n\alpha}{\beta} - \sum_{i=1}^{n} x_i = 0$
  - $\rightarrow \frac{n\alpha}{\beta} = \sum_{i=1}^{n} x_i \rightarrow \beta = \frac{n\alpha}{\sum_{i=1}^{n} x_i} \rightarrow \beta = \frac{\alpha}{\frac{1}{n}\sum_{i=1}^{n} x_i}$
- This result matches up the method of moments, but MLE needs to rely on the derivative of $\Gamma(\alpha)$, which is particularly hard.
  - $\hat{\alpha} = \frac{n\bar{x}^2}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} = \frac{\hat{\mu}_1^2}{\hat{\mu}_2 - \hat{\mu}_1^2}, \hat{\beta} = \frac{n\bar{x}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2} = \frac{\hat{\mu}_1}{\hat{\mu}_2 - \hat{\mu}_1^2}$

scipy

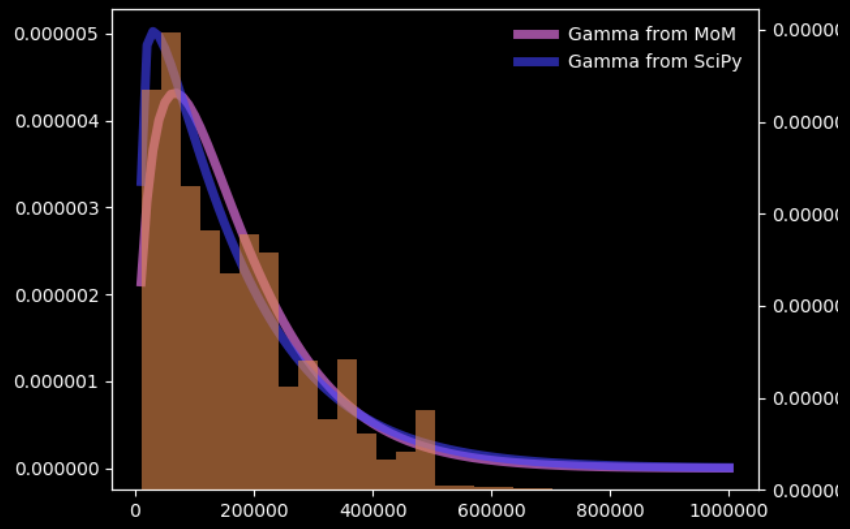
- scipy is a scientific python library providing
  - Optimization, Integration, Interpolation…
  - Algebraic equations, Differential equations…
  - Statistics, Random variables…
- Expanding the NumPy capability in computations and analyses
- SciPy provides a wrapper library whose actual implementations are in C/C++, Fortran….
- Good relations with
  - NumPy, matplotlib, pandas…..
- i.e.) SciPy provides an optimization for curve fitting, i.e. like MLE

```
Mu:  167489.2389746325
Sigma^2:  16833730579.636848
Sigma:  129744.86725738652
Alpha:  1.6664544463029376
Beta:  9.94962098165205e-06
```

Distribution Fitting by SciPy

- SciPy provides a curve fitting functionality through its optimization
  - Not an exact match: SciPy uses its own optimization algorithm and appoximations
- SciPy defines the Gamma distribution with one more parameter
  - Location parameter
    - Original : $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$
    - PDF with loc: $f(x) = \frac{\beta^\alpha}{\Gamma(\alpha)} (x-\mu)^{\alpha-1} e^{-\beta(x-\mu)}$
  - With more parameterization, curve fitting becomes better

```
Mean, Sigma :  (167489.24424801697, 129744.8639212595)
Alpha from SciPy:  1.1572769644988954
Location from SciPy:  9440.977446027482
Beta from SciPy:  7.322301167530901e-06
```



Implementation with SciPy

```python
import math
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from scipy.stats import norm, gamma
from scipy import stats

objData = pd.read_excel('../data/ucidataset-credit score/default of credit card clients.xls', skiprows=[1,2])

objResult = stats.rv_continuous.fit(norm, objData['X1'])
print("Mean, Sigma : ",objResult)

objResult = stats.rv_continuous.fit(gamma, objData['X1'])
print("Alpha from SciPy: ",objResult[0])
print("Location from SciPy: ",objResult[1])
print("Beta from SciPy: ",1/objResult[2])
```

……. calculate α, β from Method of Moments…..

```python
fig, ax1 = plt.subplots(1, 1)
ax2 = ax1.twinx()

x = np.linspace(min(objData['X1']), max(objData['X1']), 100)
ax1.plot(x, gamma.pdf(x,a=dblAlpha,scale=1/dblBeta), 'g-', lw=5, alpha=0.6, label='Gamma from MoM')
ax1.plot(x, gamma.pdf(x,objResult[0],objResult[1],objResult[2]), 'y-', lw=5, alpha=0.6, label='Gamma from
 SciPy')
ax2.hist(objData['X1'], bins=30, density=True, alpha=0.6)

ax1.legend(loc='best', frameon=False)
plt.show()
```