Statistics, Computing, and Future

Hang Li
Huawei Noah’s Ark Lab
Predicting Future
Humans Always Have Desire of Knowing Future

• Many animals have
  – Senses of space, number, and social relations
• But they do not have
  – Sense of time (i.e., cannot predict future) - Michio Kaku
• Prediction of future started at the beginning of civilization, e.g., oracle bone scripts
• Prediction of future will continue, as long as human race exists
Life can only be understood backwards; but it must be lived forwards.

人生过去可以去理解，但未来必须去经历
Predicting Future with Two Powerful Tools: Statistics and Computer
Outline

- Predicting Future with Statistics and Computer
- *Story of Blaise Pascal*
- Story of Herman Hollerith
- Statistical Machine Learning
- Story of Noah’s Ark Lab
- Summary
Blaise Pascal

- 1623-1662
- French mathematician, scientist, inventor, and philosopher
- Quote: “man is a thinking reed”
- Invented first calculator
- First studied modern probability theory with Fermat
- Developed the algorithm for calculating binomial coefficients
Pascal’s Calculator

• First mechanical calculator, for addition and subtraction, even multiplication and division
• Pascal’s father was a supervisor of taxes
• Tried 50 prototypes
• Presented first machine in 1645
• Built about 20 more machines
Problem of Points

• Game between two players
• To win a prize
• Each of them has equal chance of winning each round.
• First player who wins a certain number of rounds will collect the prize
• Suppose that game is halted during the middle
• How to divide the money fairly?
Pascal to Fermat  
Tuesday, October 27, 1654

Monsieur,

Your last letter satisfied me perfectly. I admire your method for the problem of the points, all the more because I understand it well. It is entirely yours, it has nothing in common with mine, and it reaches the same end easily. Now our harmony has begun again.

But, Monsieur, I agree with you in this, find someone elsewhere to follow you in your discoveries concerning numbers, the statements of which you were so good as to send me. For my own part, I confess that this passes me at a great distance; I am competent only to admire it and I beg you most humbly to use your earliest leisure to bring it to a conclusion. All of our gentlemen saw it on Saturday last and appreciate it most heartily. One cannot often hope for things that are so fine and so desirable. Think about it if you will, and rest assured that I am etc.

Pascal.

Paris, October 27, 1654.
Fermat and Pascal’s Solution
- in Language of Modern Math

• One player needs $r$ more rounds to win, the other player needs $s$ more rounds to win

• They should divide the prize in the ratio of

\[
\sum_{k=0}^{s-1} \binom{r+s-1}{k} / \sum_{k=s}^{r+s-1} \binom{r+s-1}{k}
\]

• Pascal first explicitly reasoned about what is known as *expectation* today

• Jacob Bernoulli published the first book on probability theory in 1713.
Probability Theory and Statistics

• Probability Theory
  – Computing possibilities of future, to predict future
  – Expectation:
    \[ E(X) = \int_{\Omega} X \, dP \]

• Statistics
  – Computing possibilities of past (as approximation of future), to predict future
  – Estimator of Expectation:
    \[ \hat{E}(X) = \frac{1}{N} \sum_{i=1}^{N} x_i \]
Pascal’s Triangle

- Triangular array of binomial coefficients
- Binomial coefficient $\binom{n}{k}$ appears in the $n$th row and $k$th item
- Starting with $\binom{0}{0} = 1$
- Recursively compute using property

\[
\binom{n}{k} = \binom{n-1}{k-1} + \binom{n-1}{k}
\]
Outline

• Predicting Future with Statistics and Computer
• Story of Blaise Pascal
• *Story of Herman Hollerith*
• Statistical Machine Learning
• Prediction in Telecommunication Networks
• Summary
Herman Hollerith

- 1860-1929
- American inventor
- Invented punched card tabulator
- Created his own company, later became International Business Machines (IBM)

Story from Prof. Bin YU
Electromechanical Punched Card Tabulator

- Punched card tabulator (穿孔制表机) - first semiautomatic data processing system
- Used in the US census in 1890, required only 6 years, v.s. previously estimated to take 13 years
Outline

• Predicting Future with Statistics and Computer
• Story of Blaise Pascal
• Story of Herman Hollerith
• *Statistical Machine Learning*
• Story of Noah’s Ark Lab
• Summary
Statistical Machine Learning Grows Dramatically Since 90’s

- Probability
- Bayesian Methods, Graphical Models, Non-Parametric Methods, Sparsity
- Decision Tree, Boosting, Kernel Methods, Deep Neural Networks

Hadoop, Spark
System
Computing
Algorithm
Computer Power Increases Exponentially
- Hans Moravec, 1999

Evolution of Computer Power/Cost

MIPS per $1000

- Billion
  - (1998)
- Million
- 10^3
- 10^2
- 10^1

First Similar Organisms
- Brain Equivalent
- Human
  - 1 MYBP
  - G4 eta 2050 (reasoning)
  - G3 eta 2040 (imagination)
- Monkey
  - 60 MYBP
  - G2 eta 2030 (adaptation)
- Mouse
  - 200 MYBP
  - G1 eta 2020 (skills)
- Lizard
  - 350 MYBP
- Guppy
  - 450 MYBP
- Worm
  - 550 MYBP
  - SRI Shakey
  - Stanford Carl
  - 1970-1980
- Bacteria
  - 3.5 Million
  - Years Before the Present
  - Hopkins Beast
  - 1960
- Manual Calculation
  - Grey Walter
  - Tortoise
  - 1950
Data Increases Exponentially
- Mary Meeker, 2006

http://www.techandinnovationdaily.com/2013/05/31/six-tech-statistics-mary-meeker/
Outline

• Predicting Future with Statistics and Computer
• Story of Blaise Pascal
• Story of Herman Hollerith
• Statistical Machine Learning
• Story of Noah’s Ark Lab
• Summary
Big Data Challenge: Prediction in Software Defined Networks (SDN)
Elephant Flow Detection

Naïve Method: Counting Using Hash Function

Key: 5 tuples of flow
Value: packet size

Challenges
- How to deal with shortage of buckets by eliminating low frequency flows (i.e., false negative)
- How to avoid miscounting due to collision of keys (i.e., false positive)
LD-Sketch: Data Structure

- Use $r$ rows of $w$ buckets with different hash functions
- Each bucket has associated array

$X$ \rightarrow h_1(x) \rightarrow h_2(x) \rightarrow h_3(x) \rightarrow h_4(x)$

- Associated array stores key-value pairs
- Registers store total errors, etc
LD-Sketch: Algorithm

• Update:
  – Get key-value pairs \((x, v)\)
  – Hash key \(x\) into a bucket in each row by different hash functions
  – Update associated array of each bucket with \((x, v)\)

• Detect:
  – Evaluate buckets
  – Output largest key-value pairs

Huang, Lee, Ld-sketch: A distributed sketching design for accurate and scalable anomaly detection in network data streams. InforCom 2014
### Update Associated Array

**Case 1:** when there is space, insert new key-value pair

<table>
<thead>
<tr>
<th>$x_3 : 3$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>3</td>
</tr>
</tbody>
</table>

**Case 2:** when key-value pair already exists, add values

<table>
<thead>
<tr>
<th>$x_3 : 2$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>8</td>
<td>5</td>
</tr>
</tbody>
</table>
Update Associated Array (cont’)

Case 3: when associated array is full, deduct minimum value, discard zero key-value pairs, insert new key-value pair

$$x_6 : 3$$

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>8</td>
<td>5</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>$x_6$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>3</td>
<td></td>
<td>2</td>
</tr>
</tbody>
</table>
Update Associated Array (cont’)

Case 4: when associated array is full and total value is larger than threshold, expand array and insert new key value pair

Case 4: when associated array is full and total value is larger than threshold, expand array and insert new key value pair.

### Array Table

<table>
<thead>
<tr>
<th>$x_6$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_7$</th>
<th>$x_5$</th>
<th>$x_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>6</td>
<td>3</td>
<td>3</td>
<td>5</td>
<td>3</td>
</tr>
</tbody>
</table>

**Total Value:** 4

**Current Value:** 4

**New Value:** 3

**New Total Value:** 7

**New Current Value:** 4
As Pseudo Tetris Game

Case 1

Case 2

Case 3

Case 4
LD-Sketch: Advantages

- Rows of buckets: reduce false positives
- Associated array: guarantee no false negative
- Space and time complexities: optimal
- Empirically significantly outperforms all baselines

Experimental Result
Elephant Flow Prediction

Predicting whether a flow is elephant flow or mouse flow by looking at the header of its first packet.

Formulation: Online Learning for Regression

\[ x_1, x_2, \ldots, x_n, x_{n+1} \]
\[ y_1, y_2, \ldots, y_n, \overline{y} \]

\[ x : \text{IP addresses, time, message length} \]
\[ y : \text{flow size} \]

Popart, Chen, Fung, and Geng, Proactive Network Routing Control System with Flow Size Prediction, to appear
Gaussian Process Regression

\[ y = f(x) + \epsilon \]

GP is a powerful tool for function approximation with non-parametric approach.

\[
\begin{bmatrix}
  y \\
  y_*
\end{bmatrix}
\sim N(0, \begin{bmatrix}
  K(X, X) + \sigma_n^2 I & K(X, x_*) \\
  K(x_*, X) & K(x_*, x_*)
\end{bmatrix})
\]

\[
\bar{y}_* = E[y_* \mid X, y, x_*] = K(x_*, X)[K(X, X) + \sigma_n^2 I]^{-1} y
\]

\[
\text{cov}(y_*) = K(x_*, x_*) - K(x_*, X)[K(X, X) + \sigma_n^2 I]^{-1} K(X, x_*)
\]

\[
\hat{y}_* = \sum_{i=1}^{n} \alpha_i K(x_*, x_i)
\]

Is also kernel method.
Example of Gaussian Process Regression

Data

\((x_1, y_1), (x_2, y_2), (x_3, y_3)\)

Predict

\(x_* \rightarrow y_* = f(x_*)?\)

\[
\begin{bmatrix}
y_1 \\ y_2 \\ y_3 \\ y_*
\end{bmatrix} \sim N
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0
\end{bmatrix},
\begin{bmatrix}
 k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \\ k_{*1} & k_{*2} & k_{*3}
\end{bmatrix}
\]

\[
k_{ij} = k(x_i, x_j) = \exp\left(-\frac{\| x_i - x_j \|^2}{2\sigma^2}\right)
\]

\[
\bar{y}_* = k_*^T (K + \sigma_n^2 \mathbf{I})^{-1} y
\]
Weight Space View: Gaussian Process Regression

\[ y_1 - f(x_1) = \varepsilon_1 \]
\[ y_2 - f(x_2) = \varepsilon_2 \]
\[ y_3 - f(x_3) = \varepsilon_3 \]

\[ \varepsilon \sim N(0, \sigma_n) \]
\[ (y - f(x)) \sim N(0, \sigma_n) \]

\[ f(x) = \phi(x)^T w \]

\[ P(y_* | x_*, X, y) \sim N(\bar{y}_*, \text{cov}(y_*)) \]
Function Space View: Gaussian Process Regression

\[ f(x) \sim N(0, \Sigma) \]

\[ \Sigma = K(X, X) + \sigma^2_n I \]

\[ f(x) = \phi(x)^T w \]

\[ P(y_* \mid x_*, X, y) \sim N(\bar{y}_*, \text{cov}(y_*)) \]
Online Gaussian Process Regression

Build subset regressor with \( m \) examples, and incrementally update with \( n, (n+1), \ldots \) examples

For sample \((x_n, y_n)\)

\[
f(x_*) = k_m(x_*)^T M_n
\]

\[
M_n = (K_{mn} K_{nm} + \sigma_n^2 K_{mm})^{-1} K_{mn} y_n
\]

For sample \((x_{(n+1)}, y_{(n+1)})\)

\[
f(x_*) = k_m(x_*)^T M_{n+1}
\]

\[
M_{n+1} = (K_{m(n+1)} K_{(n+1)m} + \sigma_n^2 K_{mm})^{-1} K_{m(n+1)} y_{n+1}
\]
Online Gaussian Process Regression

Use Woodbury matrix identity to efficiently calculate inverse of matrix

\[
\Pi_n = (K_{mn}K_{nm} + \sigma_n^2 K_{mm})^{-1}
\]

\[
\Pi_{n+1} = (K_{m(n+1)}K_{(n+1)m} + \sigma_n^2 K_{mm})^{-1}
\]

\[
= (K_{mn}K_{nm} + \sigma_n^2 K_{mm} + k_{n+1}k_{n+1}^T)^{-1}
\]

\[
= (\Pi_n + k_{n+1}k_{n+1}^T)^{-1}
\]

\[
\Pi_{n+1} = \Pi_n - \frac{\Pi_n k_{n+1}k_{n+1}^T \Pi_n}{1 + k_{n+1}^T \Pi_n k_{n+1}}
\]

Time complexity is reduced

\[
O(m^3) \rightarrow O(m^2)
\]

\[
M_{n+1} = (I - \frac{\Pi_n k_{n+1}k_{n+1}^T}{1 + k_{n+1}^T \Pi_n k_{n+1}})M_n + y_{n+1}(I - \frac{\Pi_n k_{n+1}k_{n+1}^T}{1 + k_{n+1}^T \Pi_n k_{n+1}})\Pi_n k_{n+1}
\]

\[
(A + bc^T)^{-1} = A^{-1} - A^{-1} \frac{bc^T}{1 + c^TA^{-1}b} A^{-1}
\]
**Experimental Result**

- **True Positive Rate**: (real elephant flows are correctly classified) 0.9787
- **False Positive Rate**: (mice flows are mistakenly classified as elephant flows) 0.0577
- **Computation time**: 10 µs
Outline

- Predicting Future with Statistics and Computer
- Story of Blaise Pascal
- Story of Herman Hollerith
- Statistical Machine Learning
- Story of Noah’s Ark Lab
- Summary
Big Data Era: Everything Is Digitalized
Predict Future with Statistics and Computer
Summary

• Predicting future is strong desire of humans
• Since 17 century, predicting future is carried out with two powerful tools: statistics and computer
• Statistics and computer are related at beginning
• Pascal and Hollerith are seminal figures
• Easier to predict future using statistics and computer in era of big data
• Algorithms developed at Noah’s Ark Lab
  – Elephant flow detection: LD-Sketch
  – Elephant flow prediction: Online Gaussian Process Regression
Acknowledgement

- I thank Qun Huang, Zhitang Chen, Yanhui Geng, Cheng He for helping prepare the slides
- I thank Prof. Bin Yu for sharing the story of Hollerith
Thank you!