Histograms
You’re managing the HKN lounge for next semester.

How much coffee should you buy for each day?

- Too much → waste money 😞
- Too little → under-caffeinated students 😞

What should you do?
collect data

• Count how many people get coffee in a day
  • Day 1: 37 people

• Should we just get enough coffee for 37 people?
(keep) collecting data

- Day 2: 43
- Day 3: 48
- Day 4: 41
- Day 5: 46
- Day 6: 19 (!)
- Day 7: 38
- ...

100 days later …

<table>
<thead>
<tr>
<th>37</th>
<th>43</th>
<th>48</th>
<th>41</th>
<th>46</th>
<th>19</th>
<th>28</th>
<th>35</th>
<th>34</th>
<th>38</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>32</td>
<td>32</td>
<td>23</td>
<td>23</td>
<td>33</td>
<td>35</td>
<td>39</td>
<td>34</td>
<td>28</td>
</tr>
<tr>
<td>39</td>
<td>28</td>
<td>29</td>
<td>38</td>
<td>28</td>
<td>30</td>
<td>25</td>
<td>35</td>
<td>39</td>
<td>35</td>
</tr>
<tr>
<td>31</td>
<td>28</td>
<td>25</td>
<td>26</td>
<td>15</td>
<td>31</td>
<td>28</td>
<td>32</td>
<td>40</td>
<td>21</td>
</tr>
<tr>
<td>34</td>
<td>38</td>
<td>30</td>
<td>47</td>
<td>34</td>
<td>31</td>
<td>51</td>
<td>30</td>
<td>41</td>
<td>36</td>
</tr>
<tr>
<td>33</td>
<td>51</td>
<td>22</td>
<td>25</td>
<td>29</td>
<td>50</td>
<td>32</td>
<td>39</td>
<td>25</td>
<td>37</td>
</tr>
<tr>
<td>54</td>
<td>33</td>
<td>36</td>
<td>25</td>
<td>30</td>
<td>22</td>
<td>41</td>
<td>35</td>
<td>31</td>
<td>40</td>
</tr>
<tr>
<td>30</td>
<td>33</td>
<td>27</td>
<td>36</td>
<td>27</td>
<td>34</td>
<td>24</td>
<td>41</td>
<td>37</td>
<td>29</td>
</tr>
<tr>
<td>48</td>
<td>40</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>32</td>
<td>40</td>
<td>31</td>
<td>32</td>
<td>40</td>
</tr>
<tr>
<td>31</td>
<td>33</td>
<td>32</td>
<td>38</td>
<td>37</td>
<td>41</td>
<td>37</td>
<td>39</td>
<td>38</td>
<td>42</td>
</tr>
</tbody>
</table>
visualize the data

• Staring at a list of numbers is not very illuminating

• To identify patterns in data, we should *visualize* that data in a useful way

• Idea: a histogram!
• Each bar in a histogram is a **bin**: 
  \[ x_1, x_2, x_3, \ldots \]

• Each observation goes into one bin: 
  \[ x_1 : 15 \leq d < 20 \]

• The size of each bin is the number of observations in that bin

• If we divide each count by the number of observations, we get the empirical (measured) **frequency** of each bin: 
  \[ \hat{p}_1, \hat{p}_2, \hat{p}_3, \ldots \]

• Note: \( \sum \hat{p}_k = 1 \)
visualize the data

- Remember: this histogram comes from *observed* data
- If we repeat the experiment, we might get a different histogram!
- (This is because what we have is a *sample* of the data)
visualize the data

- Remember: this histogram comes from observed data.

- If we repeat the experiment, we might get a different histogram!

- (This is because what we have is a sample of the data)
what if we collect more data?

- Hmm … this looks basically the same!
- Because we’re using the same number of bins! Each bin has more observations in it, but relative to each other, each bin is basically the same
- The frequencies of the bins aren’t changing much
so add bins!

- This looks better!
- Gives us a good sense of what the data shape looks like
... and more data!

- This looks even better!

- As we add more data points, our histogram begins to look more and more like the “true” shape of the data (we’ll get into what this means in a week or two)
how do we choose # bins?
how do we choose \# bins?

\[ k = \lceil \sqrt{n} \rceil \]

\[ k = \lceil \log_2 n \rceil + 1 \]

\[ h = \frac{3.5\hat{\sigma}}{n^{1/3}} \]

\[ k = \lceil 2n^{1/3} \rceil \]

\( k \) : \# bins

\( h \) : bin width
Intuition: the histogram estimates the “true” distribution of data using the sample of data you observe.

When given a new data point, can estimate how “likely” this data point is by looking at the frequency of the bucket the data point falls into.

- All data points that fall within the same bucket get the same estimate!

- Buckets too big: inaccurate because rare points and common points get put into the same bucket and get the same estimate.

- Buckets too small: inaccurate because sample may not have an “accurate” picture of how common that bucket is (worst case: bucket may have size zero → will estimate that the data point you just saw has no chance of happening).

- Pick a bucket size that minimizes the error of estimating any point. But how do we do this if we don’t know what the “true” data is?
cross validation

- Can use a cross-validation score.

\[ J(h) = \frac{2}{(n-1)h} - \frac{n+1}{(n-1)h}(\hat{p}_1^2 + \hat{p}_1^2 + \ldots + \hat{p}_k^2) \]
cross validation

- For a given bucket width $h$, compute $J(h)$
- Find the $h$ that minimizes $J(h)$