Alexander Lex
Mastodon: @alexlex@vis.social
http://alexander-lex.net

Provenance as a Bridge Between Data Analysis Modalities
ACKNOWLEDGEMENTS

Kiran Gadhave
Also: Jochen Görtler, Carolina Nobre, Oliver Deussen, Miriah Meyer, Jeff Phillips, Marc Streit, Nils Gehlenborg

Zach Cutler

CAREER: Enabling Reproducibility of Interactive Visual Data Analysis
WHAT IS PROVENANCE?

Provenance (from the French provenir, 'to come from/forth') is the **chronology** of the ownership, custody or location of a historical object. [Wikipedia]

In CS: a log, a **record of everything that lead to a state**
PURPOSES FOR PROVENANCE

<table>
<thead>
<tr>
<th>Purposes for Provenance</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Recall</strong></td>
<td>Maintaining or recovering memory and awareness of the current and previous states of the analysis</td>
</tr>
<tr>
<td><strong>Replication</strong></td>
<td>Reproducing the steps or workflow of a previous analysis</td>
</tr>
<tr>
<td><strong>Action recovery</strong></td>
<td>Maintaining the action history that allows undo/redo operations and branching actions during analysis</td>
</tr>
<tr>
<td><strong>Collaborative communication</strong></td>
<td>Communicating and sharing data, information, and ideas with others who are conducting the same analysis</td>
</tr>
<tr>
<td><strong>Presentation</strong></td>
<td>Communicating the insights or progression of the analysis with those who are not directly involved with the analysis themselves, such as general public, upper levels of management, or analysts focusing on other areas</td>
</tr>
<tr>
<td><strong>Meta-analysis</strong></td>
<td>Reviewing the analytic processes themselves in order to understand and improve aspects of the analysis (such as process efficiency, training efficiency, or analytic strategies)</td>
</tr>
</tbody>
</table>

Re-Application

Convert the user interactions into executable scripts.

[Xu et al 2020]

[Ragan et al 2015]
What are Modalities?

Interactive Visualization Systems

Code / Scripting
**Intuitive**

**Easy to use**

**Uses human perceptual capabilities**

---

**INTERACTIVE VISUALIZATION**

---

<table>
<thead>
<tr>
<th>Player</th>
<th>Rank</th>
<th>Team</th>
<th>Position</th>
<th>Role</th>
<th>Current_L</th>
<th>Current_P</th>
<th>Current_F</th>
<th>Age</th>
<th>Height</th>
<th>Dribbles</th>
<th>Passes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zeliqi Dzika</td>
<td>1</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Dukagjini</td>
<td>2</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Suhadri</td>
<td>3</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Suhadri</td>
<td>4</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Qerimi Dzika</td>
<td>5</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>6</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>7</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>8</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>9</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>10</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>11</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>12</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>13</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>14</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>15</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>16</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>17</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>18</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>19</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>20</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>21</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>22</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>23</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>24</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>25</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>26</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
<tr>
<td>Quini Dzika</td>
<td>27</td>
<td>Prishtina</td>
<td>Right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>right</td>
<td>25.0</td>
<td>1.86</td>
<td>2.75</td>
<td>3.90</td>
</tr>
</tbody>
</table>
Limited Expressivity
Some operations are difficult, e.g., conditional queries..
Not reusable
need to redo analysis when data changes
Not reproducible
Flexible and powerful
you basically can do anything

Reusable
if your data changes, re-run

Reproducible
everything is documented

```python
# Keep this cell
avl_df = pd.read_csv('./avalanches.csv')

# Remove NaN coordinates
avl_df = avl_df[avl_df['Coordinates'].notna()]

# Split into latitude & longitude
avl_df[['lat', 'lon']] = avl_df['Coordinates'].str.split(',', expand=True)

# Remove values outside of Utah bounds
avl_df = avl_df[avl_df['lat'].astype(float) < 42] & (avl_df['lon'].astype(float) > -180)

# Keep columns we need
avl_df = avl_df[['Date', 'Region', 'Trigger', 'lat', 'lon']]

avl_df.head()```

CODE / SCRIPTING
It’s hard
requires extensive training
reading documentation
not discoverable

Not everyone can do it

It’s time consuming

Some operations are difficult
  e.g., labeling data points

CODE / SCRIPTING: DOWNSIDES

```
# Keep this cell
avv_df = pd.read_csv('~/avalanches.csv')

# Remove NaN coordinates
avv_df = avv_df[avv_df['Coordinates']!=avv_df['Coordinates']]  

# Split into latitude & longitude
lat, lon = avv_df['Coordinates'].str.split(',', expand=True)

# Remove values outside of Utah bounds
avv_df = avv_df[(35 < avv_df['lat'].astype('float')) & (avv_df['lat'].astype('float') < 42)]
avv_df = avv_df[(114 < avv_df['lon'].astype('float')) & (avv_df['lon'].astype('float') < -100)]

# Keep columns we need
avv_df = avv_df[['Date', 'Region', 'Trigger', 'lat', 'lon']]  

avv_df.head()
```
COMPUTATIONAL NOTEBOOKS: A MIDDLE GROUND?
Yes!

Afford both scripting and interactive visualization

But visualizations are a dead end

can’t “use” interaction in code

e.g., changing a label, or filtering a value

COMPUTATIONAL NOTEBOOKS: A MIDDLE GROUND?

Jupyter Notebooks
THESIS: **BRIDGING BETWEEN CODE AND VIS IS USEFUL**

Use the best tool for each job
Provenance: record of actions that lead to a state

> Design actions such that (some) map to code operations

How?

Translate your actions to code

OR

Make code understand actions
INTERACTIVE WORKFLOWS

Specify dimensions

Brush outliers

Filter out

“Filter Outliers” Workflow
NOTABLE EXAMPLES FOR CODE SYNTHESIS

B2

[Wu et al. 2020]

Wrangler

[Kandel et al. 2011]
CHALLENGE: MAKING WORKFLOWS ROBUST
WHAT ARE SELECTIONS?
FROM SELECTIONS TO ADVANCED OPERATIONS

Filter

Label

Great Example
Interesting
Follow Up

Unassigned
Category A
Category B

Categorize

Aggregate
ROBUST SELECTIONS UNDERPIN MOST INTERACTIVE OPERATIONS
**ROBUST SELECTIONS**

**ID Based Selection:**
Elements 7, 9, 13, 18, 22

**Region-Based Selection:**
Elements that are >1.5 in $x$ and > 2 in $y$

**Semantic Selection:**
Elements in K-Means cluster centered at $[2, 3]$

**Meaningful, higher level concept:**
improves reproducibility

**Robust to changes and updates in dataset:**
enables re-usability
Intent is the user’s reason for selecting in a visualization.

Domain Specific Intent: Capture through Annotation

Pattern-Based Intent: Capture Automatically

LEVERAGING INTENT FOR ROBUST SELECTIONS
Correlation

Multivariate Optimization

Categories

Ranges

Outlier

Clusters

Correlation

Categories

Multivariate Optimization

Ranges
HOW DO WE INFERENCE INTENT?
How do we infer intent?

Selection

Predictions
K-Means
DBScan
Regression
Outlier Detection
Skyline
Decision Trees / Ranges
Categories
HOW DO WE INFER INTENT?

Selection

Predictions
- K-Means
- DBScan
- Regression
- Outlier Detection
- Skyline
- Decision Trees / Ranges
- Categories

Compare

Ranking
- Jaccard Distance
- Naive Bayes
- Classifier
- Heuristic Measures

Outlier Detection

\[ J(S, C) = \frac{|S \cap C|}{|S \cup C|} \]
HOW DO WE INFER INTENT?

Selection

Predictions
- K-Means
- DBScan
- Regression
- Outlier Detection
- Skyline
- Decision Trees / Ranges
- Categories

Ranking
- Jaccard Distance
- Naive Bayes
- Classifier
- Heuristic
- Measures

Confirming Intent & Annotation

I think this cluster...

1. Range
2. Cluster
3. Outlier
Robustness is easier in code: developers write rules, rather than lists of items.

Have to do extra work to get to the "rules" when trying to create robust workflows from interactions
HOW CAN WE INTEGRATE MODALITIES IN PRACTICE?
SPECTRUM OF VISUALIZATION SYSTEMS

**Generic Charting**
- Easy to integrate in notebooks

**Specialized Systems**
- Require standalone application
STANDALONE SYSTEMS
USING WORKFLOWS
VISUALIZATION SYSTEM

Visualization Interface

Predictions

Provenance
Interaction logs aren’t clean

Need to tidy them up
REUSING SELECTIONS ON UPDATED DATASETS

- **Changed Dataset**
- **Tracking A Selected Cluster**
- **Selected Cluster on Changed Dataset**
HUMAN REVIEWS

(a)

(b)
A workflow is a series of executable steps.

Can be abstracted and re-used in different environments.

> Bridging between our tool and Python
Interactive Visual Analysis

Computational Analysis

Library that tracks and re-executes actions

jupyter

Workflow
Database
# Using Workflow in a Computational Notebook

```python
# Installing the reapply-workflows adds a module called backend
# This module exposes the Reapply class which initializes the library
from backend import Reapply

# Here we load the reapply_workflows library.
r = Reapply()

# We add a workflow from our workflow database.
workflow = r.load_workflow("workflow1617808681620")

# Print the workflow name
print("Workflow: ", workflow.name, "\n")

# Description of the workflow and the operations in it
workflow.describe

Workflow: Deleting Cluster

<table>
<thead>
<tr>
<th>Root</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add Plot</td>
</tr>
<tr>
<td>Added brush to: X-Y</td>
</tr>
<tr>
<td>Cluster Selection</td>
</tr>
<tr>
<td>Filter: Out</td>
</tr>
</tbody>
</table>

# Prints the reapply results for all interactions, along with review status
```

# Apply the workflow to target dataset.
# apply function requires the target dataset and the label column as arguments.
res = workflow.apply(target, "Label")

# Results is an array of datasets for each interaction we grab the final one.
result_dataset = res.results[-1][\'data\']
result_dataset

This workflow has not been reviewed for all interactions. Please go to following url: [https://reapply-workflows.git](https://reapply-workflows.git)

<table>
<thead>
<tr>
<th>Label</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>57.3651</td>
<td>7.28796</td>
</tr>
<tr>
<td>5</td>
<td>4.77421</td>
<td>4.17980</td>
</tr>
<tr>
<td>8</td>
<td>8.34966</td>
<td>0.09550</td>
</tr>
<tr>
<td>9</td>
<td>8.42670</td>
<td>1.80299</td>
</tr>
<tr>
<td>10</td>
<td>4.29760</td>
<td>2.99981</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>141</td>
<td>7.35179</td>
<td>7.05215</td>
</tr>
<tr>
<td>142</td>
<td>6.62171</td>
<td>8.27311</td>
</tr>
</tbody>
</table>
BEFORE AND AFTER

150 rows × 3 columns

108 rows × 3 columns
Useful for using a standalone system with updating data using scripting after using a standalone system creating template workflows (teaching)
**SPECTRUM OF VISUALIZATION SYSTEMS**

**Generic Charting**
- Easy to integrate in notebooks

**Specialized Systems**
- Require standalone application
TRACKING PROVENANCE IN INTERACTIVE PLOTS IN NOTEBOOKS
Interactive plots are now common in notebooks (e.g., Vega-Altair)

**WHY?**

Operations such as filtering outliers or changing column labels are more efficient to do in interactive plots.
Track events in interactive visualizations

Map them to data frame operations

Operations then applied to data frame
“Listen” to all events by Vega-Altair
works for all charts that can be created with Vega-Altair
no extra effort on developer!

Custom table visualization for data frames
instead of df.head() -> use interactive table
Standard Python and Vega-Altair Code

```python
[3]:
    df = pd.read_csv("./cluster_simple_v1.csv")

    brush = alt.selection_interval(name="brs")

    alt.Chart("./cluster_simple_v1.csv", name="sp").mark_point().encode(
        x='X:Q',
        y='y:Q',
        color=alt.condition(brush, alt.value('steelblue'), alt.value('grey')),
    ).add_params(brush)

[3]:
```

Provenance Tracking Injected

Plot Generated by Vega Altair
<table>
<thead>
<tr>
<th>index</th>
<th>Name</th>
<th>Miles_per_Gallon</th>
<th>Cylinders</th>
<th>Displacement</th>
<th>Horsepower</th>
<th>Weight_in_Lbs</th>
<th>Acceleration</th>
<th>Year</th>
<th>Origin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>chevrolet chevrolet</td>
<td>18</td>
<td>8</td>
<td>307</td>
<td>130</td>
<td>3504</td>
<td>12</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>2</td>
<td>buick skylark</td>
<td>15</td>
<td>8</td>
<td>350</td>
<td>165</td>
<td>3693</td>
<td>11.5</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>3</td>
<td>plymouth satellite</td>
<td>18</td>
<td>8</td>
<td>318</td>
<td>150</td>
<td>3436</td>
<td>11</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>4</td>
<td>amc rebel ss</td>
<td>16</td>
<td>8</td>
<td>304</td>
<td>150</td>
<td>3433</td>
<td>12</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>5</td>
<td>ford torino</td>
<td>17</td>
<td>8</td>
<td>302</td>
<td>140</td>
<td>3449</td>
<td>10.5</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>6</td>
<td>ford galaxie 500</td>
<td>15</td>
<td>8</td>
<td>429</td>
<td>198</td>
<td>4341</td>
<td>10</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>7</td>
<td>chevrolet impala</td>
<td>14</td>
<td>8</td>
<td>454</td>
<td>220</td>
<td>4354</td>
<td>9</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>8</td>
<td>plymouth fury iii</td>
<td>14</td>
<td>8</td>
<td>440</td>
<td>215</td>
<td>4312</td>
<td>8.5</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>9</td>
<td>pontiac catalina</td>
<td>14</td>
<td>8</td>
<td>455</td>
<td>225</td>
<td>4425</td>
<td>10</td>
<td>0</td>
<td>USA</td>
</tr>
<tr>
<td>10</td>
<td>amc ambassador</td>
<td>15</td>
<td>8</td>
<td>390</td>
<td>190</td>
<td>3850</td>
<td>8.5</td>
<td>0</td>
<td>USA</td>
</tr>
</tbody>
</table>
Editing column names, cells
Sorting rows/columns
Dropping rows/columns
Filtering (in/out) items (with intent predictions)
Labeling items
Categorizing items
Grouping / Aggregating items
Provenance enables **undo/redo** in notebook
Branching enables **alternative explorations**
Minimal invasiveness makes it **easy to adopt**
Fully **reusable**, just like code

**User interaction when useful, use coding when more efficient!**
Provenance can be used to bridge analysis modalities!

**Translation** to meaningful operation isn’t always easy - more work needed!

Low-level charts are ripe target for **integrating interactivity in notebooks**
Thanks to: Kiran Gadhave, Zach Cutler, Carolina Nobre, Marc Streit, Jochen Görtler, Oliver Deussen, Miriah Meyer, Jeff Phillips, Samuel Gratzl, Holger Stitz, Nils Gehlenborg, Hendrik Strobelt, Romain Vuillemot, Hanspeter Pfister, and many others!

Alexander Lex
Mastodon: @alexlex@vis.social
http://alexander-lex.net