Reusing Interactive Analysis Workflows

Kiran Gadhave, Zach Cutler, Alexander Lex
University of Utah
Interactive Visual Analysis

Computational Analysis

< / >
Interactive Visual Analysis
Interactive Visual Analysis

• Intuitive
Interactive Visual Analysis

- Intuitive
- Uses human perceptual capabilities
Interactive Visual Analysis

• Intuitive
• Uses human perceptual capabilities

• Need to redo the analysis when the datasets update
Computational Analysis
Computational Analysis

• Flexible
Computational Analysis

• Flexible

• Reusable
Computational Analysis

- Flexible
- Reusable

- Have to know how to program
Computational Analysis

- Flexible
- Reusable

- Have to know how to program
- Time consuming
Certain tasks are easier with interactive visualizations
Certain tasks are easier with interactive visualizations

Labeling
Aggregation
Filtering
Categorization
In a computational notebook
In a computational notebook

```python
sel = dataframe[:, ...]
```
In a computational notebook

```python
sel = dataframe[:, ...]

sel.add_column(A)
```
In a computational notebook

```
sel = dataframe[:, ...

sel.add_column(A)
```

```
sel = dataframe[:, ...

sel.add_column(B)
```

```
sel = dataframe[:, ...

sel.add_column(C)
```
Depending on the query predicate, this can get real complex.
whereas with an interactive visualization
whereas with an interactive visualization
whereas with an interactive visualization
But what if we want to reuse our analysis?
But what if we want to reuse our analysis?

Functions can be parameterized and reused

```python
def add_category(dataframe, col):
    selection = dataframe[:, ...]
    selection.add_column(col)
```
But what if we want to reuse our analysis?

Functions can be parameterized and reused

```python
def add_category(dataframe, col):
    selection = dataframe[:, ...]
    selection.add_column(col)
```

Interactions need to be repeated
Can we make visual analysis reusable?
Contribution

Capture analysis provenance
Contribution

Capture analysis provenance
and curate reusable workflows
Contribution

Reusable Workflows

Reapply the workflows on updated datasets

Apply the workflow in a different environment
Workflows
Workflows
Workflows

Sequence of tasks
Workflows

Computational Environment

```r
gapminder %>%
  select(country, lifeExp, gdpPerCap) %>%
  filter(country == 'India')
```
Workflows

Interactive Visualizations

?
Workflow Creation
Workflow Creation

Explicit Modeling          Process-based
Explicit Modeling

Specify each task individually and specify their sequence

Load Data → Select Range → Filter Selection → Resize Plot

Similar to programming

Does not support rapid exploration
Explicit Modeling

VisTrails

[VisTrails Image]

VisFlow

[VisFlow Image]

[Bavoil et. al., 2005]  

[Yu and Silva, 2017]
Process-based

After finding: Leverage analysis provenance to curate a workflow

Explore the data

Captured analysis  Curated workflow
Process-based

Vistories

However, those living in the African continent still remain relatively poor and sick.

From 1800 to 2015, overall we can see that as people got richer, they also started living longer.

[Gratzl et. al., 2016]
Process-based

Easy & Natural

Freeform unencumbered exploration
Capturing Workflows

- Add Scatterplot
- Select Points
- Filter Out Selection
- Resize plot

Captured Analysis
Capturing Workflows

Captured Analysis
- Add Scatterplot
- Select Points
- Filter Out Selection
- Resize plot

Filter Outliers Workflow
- Add Scatterplot
- Select Points
- Filter Out Selection
Such workflows enable reproducibility
Such workflows enable reproducibility
Such workflows enable reproducibility

But what if the dataset changes?
Capturing Workflows

Captured Analysis
- Add Scatterplot
- Select Points
- Filter Out Selection
- Resize plot

Filter Outliers Workflow
- Add Scatterplot
- Select Points
- Filter Out Selection

Apply workflow
Capturing Workflows

Captured Analysis
- Add Scatterplot
- Select Points
- Filter Out Selection
- Resize plot

Filter Outliers Workflow
- Add Scatterplot
- Select Points
- Filter Out Selection

Apply workflow
Why?
Why?

The way selections are captured in the provenance
ID Based Selection:
Selected Elements: 7, 9, 13, 18, 22
ID Based Selection:
Selected Elements: 7, 9, 13, 18, 22

Range Based Selection:
Rectangular area from (1,2) to (5,7)
ID Based Selection:
Selected Elements: 7, 9, 13, 18, 22

Range Based Selection:
Rectangular area from (1,2) to (5,7)

Semantic Selection:
Elements in K-Means cluster centered at [2, 3]
Meaningful, higher level concept:
improves reproducibility

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

**Range Based Selection:**
Rectangular area from (1,2) to (5,7)

**Semantic Selection:**
Elements in K-Means cluster centered at [2, 3]
ID Based Selection:
Selected Elements: 7, 9, 13, 18, 22

Range Based Selection:
Rectangular area from (1,2) to (5,7)

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
Predicting intent behind selections in scatterplot visualizations

Kiran Gadhave1, Jochen Görtler2, Zach Cutler1, Carolina Nobre3, Oliver Deussen2, Miriah Meyer1, Jeff M. Phillips1 and Alexander Lex1

Abstract
Predicting and capturing an analyst’s intent behind a selection in a data visualization is valuable in two scenarios. First, a successful prediction of a pattern an analyst intended to select can be used to auto-complete a partial selection which, in turn, can improve the correctness of the selection. Second, knowing the intent behind a selection can be used to improve recall and reproducibility. In this paper, we introduce methods to infer analyst’s intents behind selections in data visualizations, such as scatterplots. We describe intents based on patterns in the data, and identify algorithms that can capture these patterns. Upon an interactive selection, we compare the selected items with the results of a large set of computed patterns, and use various ranking approaches to identify the best pattern for an analyst’s selection. We store annotations and the metadata to reconstruct a selection, such as the type of algorithm and its parameterization, in a provenance graph. We present a prototype system that implements these methods for tabular data and scatterplots. Analysts can select a prediction to auto-complete partial selections and to seamlessly log their intents. We discuss implications of our approach for reproducibility and reuse of analysis workflows. We evaluate our approach in a crowd-sourced study, where we show that auto-completing selection improves accuracy, and that we can accurately capture pattern-based intent.

Keywords
Provenance, reproducibility, intent, brushing, selections

Introduction
When experts interact with a visual analysis system, they are frequently guided by a domain-specific analysis question, such as identifying a gene that could be a drug target. To answer this question, they execute a series of intermediate tasks, such as selecting a set of correlated items for detailed analysis. In contrast to the high-level goal of answering a domain-specific question, these intermediate tasks are based on patterns in the data, which are distinct from higher level intents in that they are free of context and based solely on the data. They are also distinct from low-level intents, such as hovering over an item to read its label. In this paper, we introduce methods to infer these pattern-based intents for brushes in scatterplots. We define pattern-based intents as the reasoning behind selections based on statistical patterns.
Capturing Intent

Selection
Capturing Intent

Selection

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

Selection

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

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

Selection

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

Ranking
- Jaccard Distance
- Naive Bayes
- Classifier
- Heuristic Measures
Capturing 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...
Capturing Reusable Workflow
Capturing Reusable Workflow

Add Scatterplot
Select Points
Refine Selection to Select Outliers
Filter Out Selection

Captured Analysis
Capturing Reusable Workflow

- Add Scatterplot
- Select Points
- Refine Selection to Select Outliers
- Filter Out Selection

Captured Analysis

Algorithms:
- DBScan
- Local Outlier Factor
...
Capturing Reusable Workflow

Captured Analysis

- Add Scatterplot
- Select Points
- Refine Selection to Select Outliers
- Filter Out Selection

Semantic Filter Outlier Workflow

- Add Scatterplot
- Select Outliers
- Filter Out Selection
Capturing Reusable Workflow

Captured Analysis

Semantic Filter Outlier Workflow

- Add Scatterplot
- Select Points
- Refine Selection to Select Outliers
- Filter Out Selection
- Add Scatterplot
- Select Outliers
- Filter Out Selection

Updated Dataset
Capturing Reusable Workflow

Captured Analysis
Add Scatterplot
Select Points
Refine Selection to Select Outliers
Filter Out Selection

Semantic Filter Outlier Workflow
Add Scatterplot
Select Outliers
Filter Out Selection

Apply Semantic Workflow
Add Scatterplot
Select Outliers
Filter Out Selection

Kiran Gadhave
Example: Selecting a cluster
Example: Selecting a cluster

Dataset updates
Comparing the selections
Final selection after automatic reapplication
Reviewing applied workflows

Original cluster selection

Diagram showing scatter plots and cluster selections.
Reviewing applied workflows

Original cluster selection

Broken cluster
Reviewing applied workflows

Original cluster selection

Broken cluster
Demo
Bridging between environments

Data analysis rarely takes place in a single tool
Bridging between environments

Data analysis rarely takes place in a single tool

Raw data exploration
Wrangling
Exploratory analysis
Reporting

Tableau/PowerBI
Jupyter/R notebooks

SQL
Spreadsheets
Bridging between environments

Exporting the transformed data is difficult
Bridging between environments

Exporting the transformed data is difficult

Need to redo parts of analysis when switching tools
Contribution

Reuse workflows in a different environment
Reusing workflows in Jupyter

Captured analysis
Reusing workflows in Jupyter

Captured analysis

Curated workflow
Reusing workflows in Jupyter

Workflow Database

A → B → C → D → E → F → G

B → C → F

B → G

B → H

B → L

I → T → B
Reusing workflows in Jupyter

Workflow Database

import workflow
apply workflow
further analysis
Reapply Library
Reapply Library

Python Library
Reapply Library

Python Library

Load workflows
Reapply Library

Python Library

Load workflows

Core logic for capturing and applying workflows
Reapply Library

Python Library

Load workflows

Core logic for capturing and applying workflows

Apply to pandas dataframe
Reapply Library

Python Library

Load workflows

Core logic for capturing and applying workflows

Apply to pandas dataframe

Proof-of-concept
Demo

# This module exposes the Reapply class
from reapply_workflows import Reapply
# This module exposes the Reapply class
from reapply_workflows import Reapply
Demo

# Initialize the Reapply library
reapply = Reapply()

# Load the Covid Dataset project
project = reapply.load("Covid OWID")
project.list_workflows()

Categorize outliers - 1638475878304
# This module exposes the Reapply class
from reapply_workflows import Reapply

# Initialize the Reapply library
reapply = Reapply()

# Load the Covid Dataset project
project = reapply.load("Covid OWID")
project.list_workflows()

Categorize outliers — 1638475878304
# Get the desired workflow

```python
wf = project.get_workflow("1638475878304")
```

# Description of the options in the workflow

```python
wf.describe()
```

Categorize outliers
- Root
  - Adding scatterplot for new_cases_per_million-new_deaths_per_million
    - Apply Outlier selection
      - Filter In
        - Add Brush
          - Categorize Selections
            - Add Brush
              - Update Brush
                - Categorize Selections
                  - Add Brush
                    - Categorize Selections
```

Kiran Gadhave
# Get the desired workflow
```python
wf = project.get_workflow("1638475878304")
```

# Description of the options in the workflow
```python
wf.describe()
```

Categorize outliers
| Root
|---| Adding scatterplot for new_cases_per_million-new_deaths_per_million
|---| Apply Outlier selection
|---| Filter In
|---| Add Brush
|---| Categorize Selections
|---| Add Brush
|---| Update Brush
|---| Categorize Selections
|---| Add Brush
|---| Categorize Selections
Demo

```python
# Get the desired workflow
wf = project.get_workflow("1638475878304")

# Description of the options in the workflow
wf.describe()
```

Categorize outliers
| Root
  | Adding scatterplot for new_cases_per_million-new_deaths_per_million
  | Apply Outlier selection
  | Filter In
  | Add Brush
  | Categorize Selections
  | Add Brush
  | Update Brush
  | Categorize Selections
  | Add Brush
  | Categorize Selections

January 2021

- UNASSIGNED
- HIGH DEATHS - HIGH CASES
- LOW DEATH - HIGH CASES
- HIGH DEATHS - LOW CASES

D | New Deaths (per million)
C | New Cases (per million)
# This module exposes the Reapply class
from reapply_workflows import Reapply

# Initialize the Reapply library
reapply = Reapply()

# Load the Covid Dataset project
project = reapply.load("Covid OWID")
project.list_workflows()

Categorize outliers - 1638475878304

# Get the desired workflow
wf = project.get_workflow("1638475878304")

# Description of the options in the workflow
wf.describe()

Categorize outliers
| Root
  | Adding scatterplot for new_cases_per_million-new_deaths_per_million
  | Apply Outlier selection
  | Filter In
  | Add Brush
  | Categorize Selections
  | Add Brush
  | Update Brush
  | Categorize Selections
  | Add Brush
  | Categorize Selections
Demo

```python
# Apply the workflow to three versions of the dataset
results_jan_2021 = wf.apply(jan_2021, "location")
results_dec_2020 = wf.apply(dec_2020, "location")
results_june_2021 = wf.apply(june_2021, "location")
```
# This module exposes the Reapply class
from reapply_workflows import Reapply

# Initialize the Reapply library
reapply = Reapply()

# Load the Covid Dataset project
project = reapply.load("Covid OWID")
project.list_workflows()

Categorize outliers - 1638475878304

# Get the desired workflow
wf = project.get_workflow("1638475878304")

# Description of the options in the workflow
wf.describe()

Categorize outliers
| Root
  |     Adding scatterplot for new_cases_per_million-new_deaths_per_million
  |     | Apply Outlier selection
  |     |     Filter In
  |     |     | Add Brush
  |     |     | Categorize Selections
  |     |     | Add Brush
  |     |     | Update Brush
  |     |     | Categorize Selections
  |     |     | Add Brush
  |     |     | Categorize Selections
Demo

# This module exposes the Reapply class
from reapply_workflows import Reapply

# Initialize the Reapply library
reapply = Reapply()

# Load the Covid Dataset project
project = reapply.load("Covid OWID")
project.list_workflows()

Categorize outliers - 1638475878304

# Get the desired workflow
wf = project.get_workflow("1638475878304")

# Description of the options in the workflow
wf.describe()

Categorize outliers

| Root
| ---| Adding scatterplot for new_cases_per_million-new_deaths_per_million
|  | ---| Apply Outlier selection
|  |   | Filter In
|  |   | Add Brush
|  | ---| Categorize Selections
|  |   | Update Brush
|  |   | Categorize Selections
|  |   | Add Brush
|  | ---| Categorize Selections

# Apply the workflow to three versions of the dataset
results_jan_2021 = wf.apply(jan_2021, "location")
results_dec_2020 = wf.apply(dec_2020, "location")
results_june_2021 = wf.apply(june_2021, "location")
Evaluation
Evaluation

Usage Scenarios
Evaluation

Usage Scenarios

Covid - Our World in Data
Evaluation

Usage Scenarios

Covid - Our World in Data

Gapminder Public Health
Evaluation
Expert Feedback

4 data practitioners
Evaluation

Expert Feedback

Interview

Introduction to the techniques

Demo

Feedback
Evaluation

Expert Feedback

About curating workflows from provenance

I like this, because it’s much more natural.

About reusing the workflows on updated datasets

I definitely think it will be applicable because most of the time, we actually don’t inherently change the method itself … so I definitely can see this to be helpful

About reusing the workflows in a computational environment

Great to be able to **click on the Select 53 points**, and then **see the all code**, you know, to that would select the 53 points.
Future Work
Future Work

Workflows as templates
Future Work

Workflows as templates

Reapplying captured workflows on unrelated datasets
Future Work

Workflows as templates

Reapplying captured workflows on unrelated datasets

Automate repetitive data preprocessing steps
Future Work

**Workflows as templates**

Reapplying captured workflows on unrelated datasets

Automate repetitive data preprocessing steps

Training using workflows curated by experts
Future Work

Integration with interactive visualizations in the notebook environment
Future Work

Integration with interactive visualizations in the notebook environment

Libraries like Altair support interactive visualizations in notebook environment.

Typically the interactions cannot manipulate the data.
Future Work

Integration with interactive visualizations in the notebook environment

B2 — introduces techniques to coordinate interactive visualizations and code cells.

Data queries (or selections) act as the bridge between interactive visualizations and the code.

[Wu et. al., 2020]
Future Work

Integration with interactive visualizations in the notebook environment

Integrate interactive visualizations with the notebook

Semantic analysis provenance as a shared abstraction
Thank You

Questions

We thank Derya Akbaba and Jack Wilburn from VDL for help with the expert interviews.

Supported by National Science Foundation (IIS 1751238)