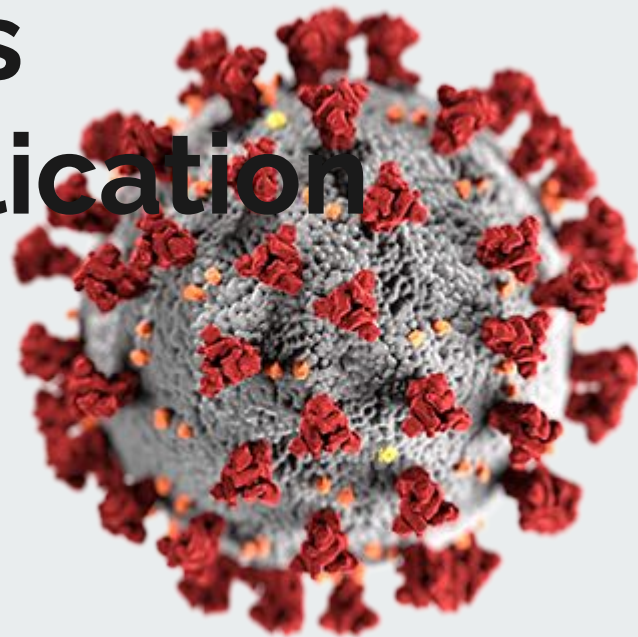




COVID-19 Analysis Visualization Application

Data 228
Group Alright

Zhe Li
Jingqi Guan
Yun Miao
Ting Deng





Project Overview:

- Visualize the Covid-19 data across over time;
- Analysis the Covid-19 trend in different condition;
- Utilize Amazon Web Service to host and secure our application and data;
- Spread correct and secured Covid-19 data and proper interpretation to public;
- Provide deep insights to our target users, on both current situation and predicting the future world with Covid-19;



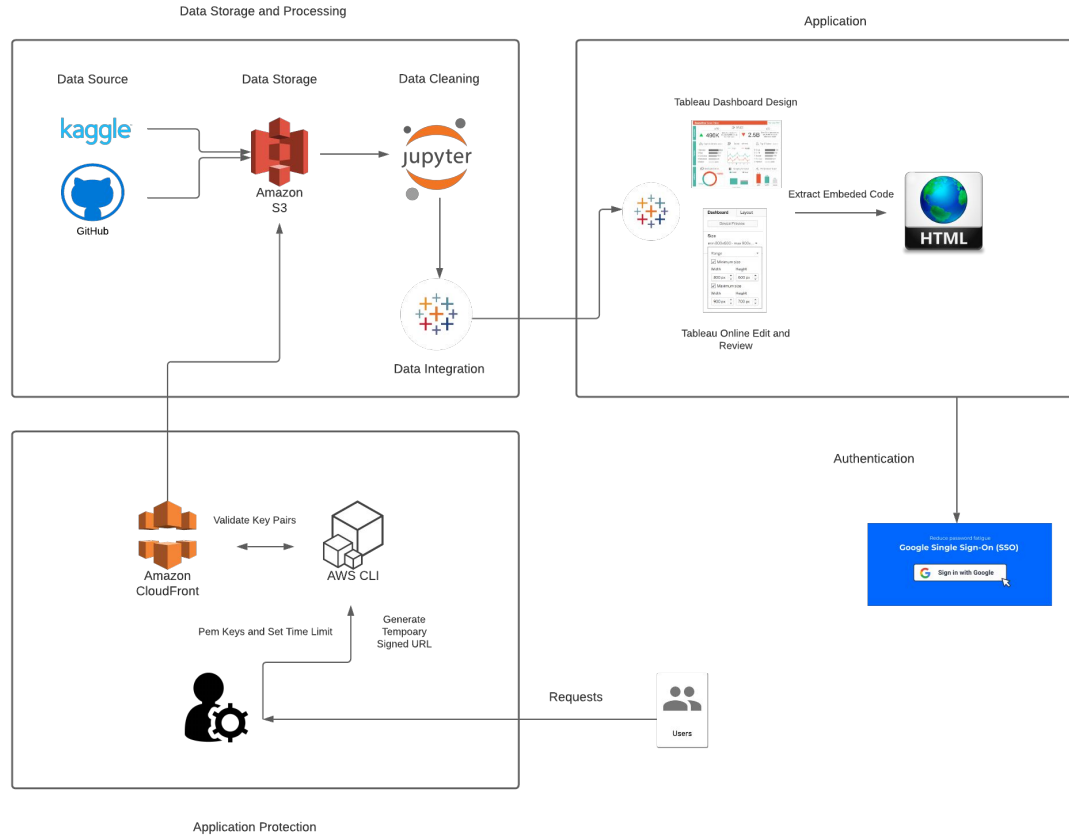
Introduction

- The Covid-19 pandemic has been with us for more than 2 years, and it has developed numerous contagious variants. It cost us excess deaths as a true toll for late response. Covid-19 will likely be with us forever, according to WHO, and it is foreseeable that it will have more variants in the future with higher reproductive numbers and death rate.
- We performed an explanatory analysis on the data published by John Hopkins University, including the reported Covid-19 case on each county, and the vaccination rate subject to each state in the US.



Target Users

- Travelers
 - Increase knowledge of the US Covid Trends
 - Smartly decide the travel destination
- Government and Policy Makers;
- Health service provider;
- US residence;



Development Process



Data Description

- **Covid-US-county dataset** - The data is provided by the [Johns Hopkins University](#) through their excellent [github repo](#). The earliest recorded cases are from 2020-01-22
- **US_county dataset** - Demographic information on the US county level based on the (most recent) 2014-18 release of the American Community Survey
- **US_state_Vaccinations dataset** - Data collected from [Our World in Data's github repository](#) which collects US vaccination data from the [CDC](#). Includes vaccination progress of the US as a whole.

Data processing

Checking the missing values on the right

```
In [9]: ##restructure df with selected features
df_vacci_new = df_vacci.loc[:,["date","location","total_vaccinations"]]
df_vacci_new = df_vacci_new.dropna(axis=0)
df_vacci_new.head()
```

```
Out[9]:
```

	date	location	total_vaccinations
0	2021-01-12	Alabama	78134.0
1	2021-01-13	Alabama	84040.0
2	2021-01-14	Alabama	92300.0
3	2021-01-15	Alabama	100567.0
7	2021-01-19	Alabama	130795.0

```
In [11]: df_popu_new = df_popu.loc[:,["county","state","population","median_age"]]
df_popu_new = df_popu_new.dropna(axis=0)
df_popu_new.head()
```

```
Out[11]:
```

	county	state	population	median_age
0	Autauga County	Alabama	55200	37.8
1	Baldwin County	Alabama	208107	42.8
2	Barbour County	Alabama	25782	39.9
3	Bibb County	Alabama	22527	39.9
4	Blount County	Alabama	57645	40.8

```
In [5]: # %% Check missing values in vacci file
missing = df_vacci.isna().sum().apply(lambda x: x/df_vacci.shape[0] * 100)
print(missing[missing > 0].sort_values(ascending = False))
```

```
people_fully_vaccinated_per_hundred    10.039644
people_vaccinated_per_hundred           9.653504
distributed_per_hundred                 9.596870
total_vaccinations_per_hundred          9.529939
daily_vaccinations_per_million          6.435669
daily_vaccinations_raw                   6.281213
people_fully_vaccinated                   5.405962
people_vaccinated                        4.994079
total_distributed                        4.937445
share_doses_used                         4.937445
total_vaccinations                       3.634866
daily_vaccinations                       0.334655
dtype: float64
```

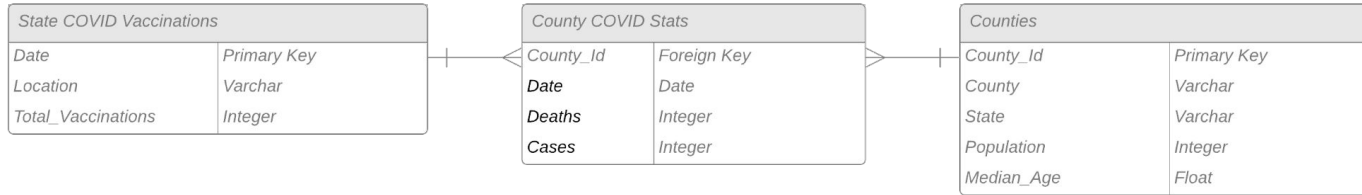
```
In [6]: # %% Check missing values in population file
missing = df_popu.isna().sum().apply(lambda x: x/df_popu.shape[0] * 100)
print(missing[missing > 0].sort_values(ascending = False))
```

```
state_code    2.453416
dtype: float64
```

```
In [7]: # %% Check missing values in covid file
issing = df_covid.isna().sum().apply(lambda x: x/df_covid.shape[0] * 100)
rint(missing[missing > 0].sort_values(ascending = False))
```

```
tate_code    2.663076
ips          0.299222
ounty        0.179533
type: float64
```

Restructure the data with selected features on the left



Edit Relationship ×

How do relationships differ from joins? [Learn more](#)

covidnew.csv		vaccnew.csv
Date	=	Date (Vaccinew.Csv)
State (Covidnew.Csv)	=	Location
<div>⊕ Add more fields</div>		
<div>> Performance Options</div>		

Integrate relational data from different source using tableau

Cloudfront and S3 - protection for data

Path pattern [Info](#)

Default (*)

Origin and origin groups

data228application.s3.us-west-1.amazonaws.com ▼

Compress objects automatically [Info](#)

☐ No

☒ Yes

Viewer

Viewer protocol policy

☐ HTTP and HTTPS

☒ Redirect HTTP to HTTPS

☐ HTTPS only

Block public access (bucket settings)

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access. If you require some level of public access to your buckets or objects, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects, turn off Block all public access and configure individual settings below to suit your specific storage use cases. [Learn more](#) [↗](#)

Edit

Block *all* public access

✔ On

Block public access to buckets and objects granted through *new* access control lists (ACLs)

✔ On

Block public access to buckets and objects granted through *any* access control lists (ACLs)

✔ On

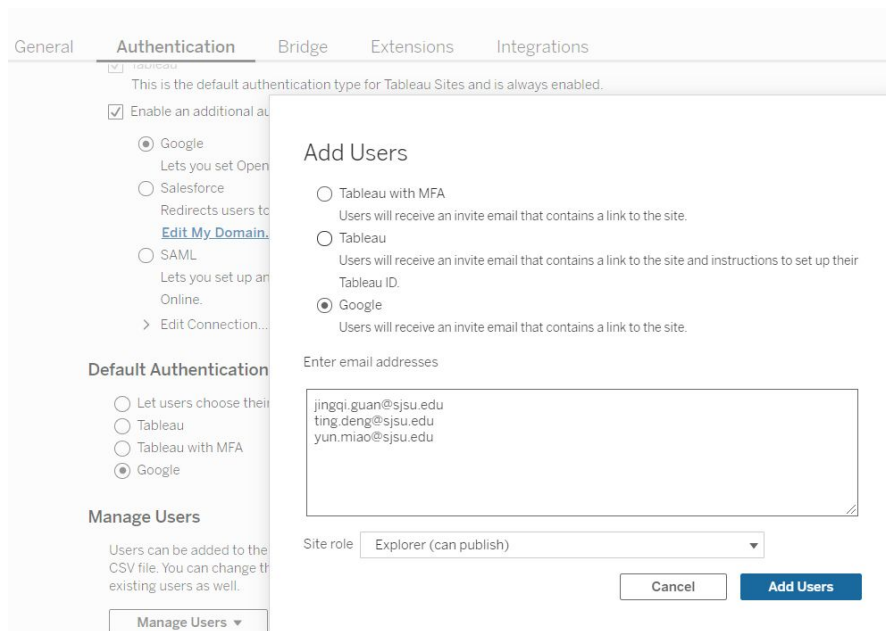
Block public access to buckets and objects granted through *new* public bucket or access point policies

✔ On

Block public and cross-account access to buckets and objects through *any* public bucket or access point policies

✔ On

SSO - protection for production



The screenshot shows the 'Authentication' tab in the Tableau Site Settings. The 'General' tab is selected, and the 'Authentication' sub-tab is active. The 'Default Authentication' section shows 'Google' as the selected option. The 'Add Users' dialog is open, showing the 'Add Users' tab. The 'Add Users' dialog has a text area for email addresses containing three email addresses: jingqi.guan@sjsu.edu, ting.deng@sjsu.edu, and yun.miao@sjsu.edu. The 'Site role' dropdown is set to 'Explorer (can publish)'. The 'Add Users' button is highlighted in blue.

General **Authentication** Bridge Extensions Integrations

☒ Tableau

This is the default authentication type for Tableau Sites and is always enabled.

☒ Enable an additional authentication type

☒ Google
Lets you set OpenID Connect as the default authentication type.

☐ Salesforce
Redirects users to a Salesforce login page. See [Edit My Domain](#) for more information.

☐ SAML
Lets you set up an SAML connection to an external identity provider.

> Edit Connection...

Default Authentication

☐ Let users choose their authentication type

☐ Tableau

☐ Tableau with MFA

☒ Google

Add Users

☐ Tableau with MFA
Users will receive an invite email that contains a link to the site.

☐ Tableau
Users will receive an invite email that contains a link to the site and instructions to set up their Tableau ID.

☒ Google
Users will receive an invite email that contains a link to the site.

Enter email addresses

jingqi.guan@sjsu.edu
ting.deng@sjsu.edu
yun.miao@sjsu.edu

Site role: Explorer (can publish)

Cancel Add Users

Manage Users

AWS CLI and Signed URLs - protection for deployment

Restrict viewer access

If you restrict viewer access, viewers must use CloudFront signed URLs or signed cookies to access your content.

- ☐ No
- ☒ Yes

Trusted authorization type

- ☒ Trusted key groups (recommended)
- ☐ Trusted signer

Add key groups

Choose the key groups

demo X

[Create key group](#)

```
sign
--url <value>
--key-pair-id <value>
--private-key <value>
--date-less-than <value>
[--date-greater-than <value>]
[--ip-address <value>]
```

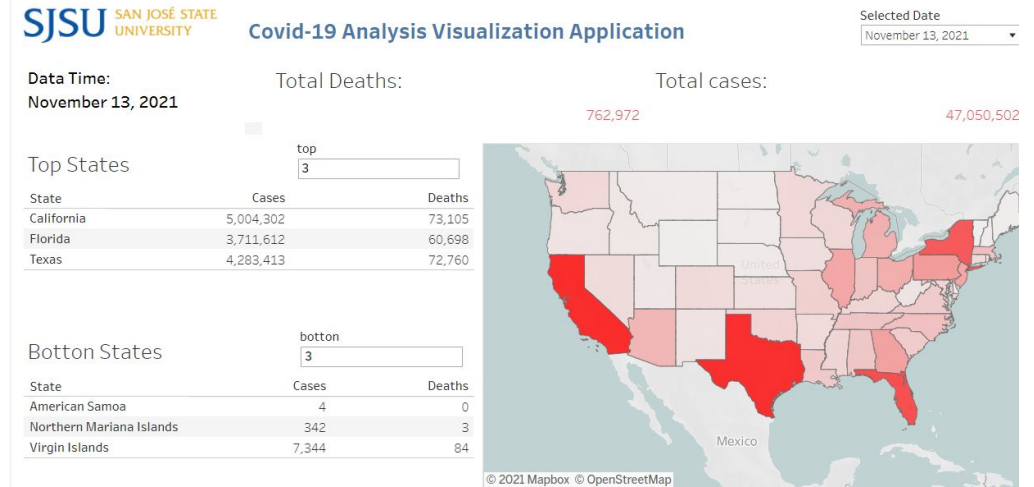


Application demo

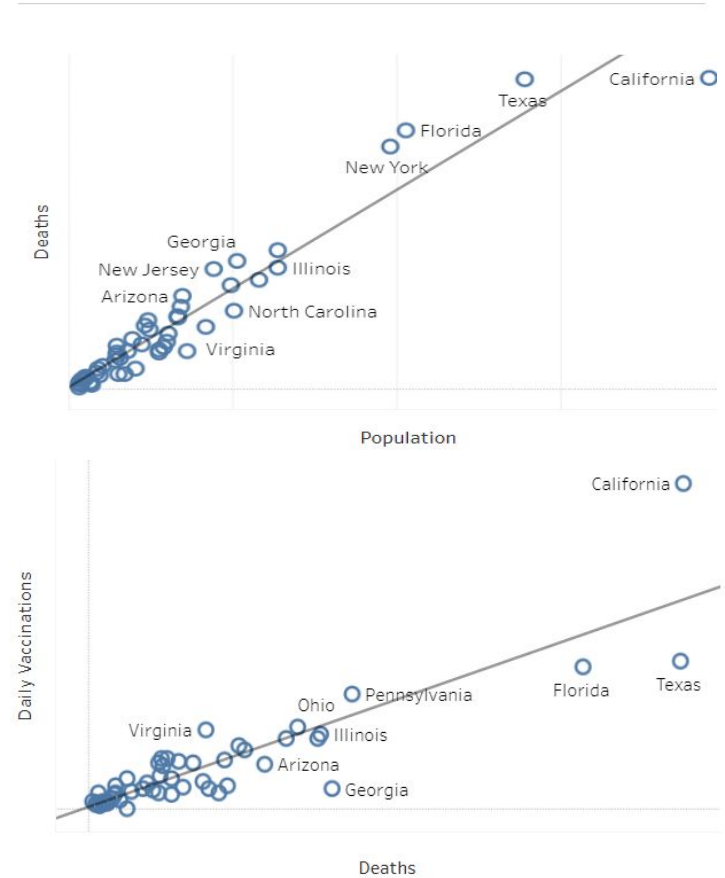
The following is the signed URLs which can last for 7 days

https://d1jfi0hxjl7gxv.cloudfront.net/?Expires=1638230400&Signature=S1QurlF4U75PzDx~3Azq0nZBwh6yzleVcRUZtPvWTiANRC-Bg52PzKUOeiVBfBfsRXfX2SXJXvieljGr3LtTJ5PFsd3QqObHNXje5vuLebLFE1PX43YmBXluLrnusjuUaF-lckVq8cptroNW5E-c8y7bAmPG-Pp~e3YrIGR3b-xQGIWgLo1REE7C4VUiioFHkqDmEzoQ1QnN61uUBGPgDxND6bW9Vf2tbl3X--6qor~ICboeOipXL2wZR7z2vQ164DGGoY7JS1J-FZN5LM3AktBYpsZ4oy0xfOr5Gnfd6PX55q594mb~2zMgz5MgXqHk22LN75ybHx0DkKeKe8LJw__&Key-Pair-Id=K2CTPBXSH0X97Y

Visualize Covid data across time



Analyze trend in different condition





What did we learn

- Data Process and Integration
- Data Storage and Security
- Application Development Process
- Application Deployment and Security
- Visualization Techniques
- Project Management



Future work

- Gathering more data: worth for cloud computing on AWS
- Collecting other related data: more innovative findings
- Applying Machine learning and algorithms: SIR models to predict the trend of Covid-19 and appearance of Delta species.



Thank You!!!
Any Questions?