

# **ДОСТУПНОСТЬ ОБРАЗОВАТЕЛЬНЫХ ПРОЕКТОВ В УСЛОВИЯХ ЦИФРОВИЗАЦИИ**

Ксения Усович,  
Университет Калифорнии, Беркли

# Обо мне



- Master's Student in Information & Data Science, UC Berkeley
- Python Instructor
- External Adoption Team Lead, Division of Computing, Data Science, and Society
  
- Студент Магистратуры, Информационные Науки, Университет Калифорнии, Беркли (UC Berkeley)
- Преподаватель Python
- Руководитель группы по внешнему распространению программ по Информационным наукам

Где учиться?



# Обзор

- JupyterHub удобен в использовании, но не доступен для людей вне Беркли

Berkeley  
UNIVERSITY OF CALIFORNIA

Operated by the **Division of Computing, Data Science, and Society**

Log in to continue

Welcome to the University of  
California, Berkeley **DataHub**.

 jupyter

 R Studio

# Обзор программ

The screenshot displays the Anaconda Navigator web interface. At the top, the logo "ANACONDA.NAVIGATOR" is visible on the left, and "Upgrade Now" and "Sign in" buttons are on the right. A left sidebar contains navigation options: Home, Environments, Learning, and Community. Below these is a "ANACONDA NUCLEUS" banner with a "Join Now" button, followed by links for "Documentation" and "Anaconda Blog", and social media icons for Twitter, YouTube, and GitHub. The main content area is titled "Applications on" with a dropdown menu set to "sr\_validation" and a "Channels" filter. It features a grid of eight application cards, each with a logo, name, version, description, and a "Launch" button. The applications are: Datalore (Online Data Analysis Tool), IBM Watson Studio Cloud (tools for data analysis and machine learning), JupyterLab (version 3.2.1, extensible environment), Jupyter Notebook (version 6.4.6, web-based environment), PyCharm Community (IDE by JetBrains for Python), Glueviz (version 1.0.0, multidimensional data visualization), Orange 3 (version 3.26.0, component-based data mining), and PyCharm Professional (full-fledged IDE for Scientific and Web Python).

ANACONDA.NAVIGATOR Upgrade Now Sign in

Home Environments Learning Community

ANACONDA NUCLEUS Back up your environments in Nucleus for free Join Now

Easily back up, port, and restore any environment

Documentation Anaconda Blog

Twitter YouTube GitHub

Applications on sr\_validation Channels

**DL** Datalore Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team. Launch

**IBM** IBM Watson Studio Cloud IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling. Launch

**lab** JupyterLab 3.2.1 An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture. Launch

**jupyter** Jupyter Notebook 6.4.6 Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis. Launch

**PC** PyCharm Community An IDE by JetBrains for pure Python development. Supports code completion, listing, and debugging.

**Glueviz** 1.0.0 Multidimensional data visualization across files. Explore relationships within and among related datasets.

**Orange 3** 3.26.0 Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

**PC** PyCharm Professional A Full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.

Anaconda

# Обзор программ (cont'd)

Jakub Jurovych Exploratory analysis

Running 2 cells Stop execution

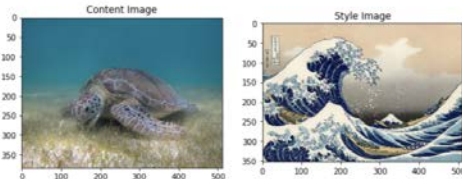
```
# Set up some global values here
content_path = './images/Green_Sea_Turtle_grazing_seagrass.jpg'
style_path = './images/The_Great_Wave_off_Kanagawa.jpg'
```

### Visualize the input

```
plt.figure(figsize=(10,10))
content = load_img(content_path).astype('uint8')
style = load_img(style_path).astype('uint8')

plt.subplot(1, 2, 1)
imshow(content, 'Content Image')
cl
```

content	variable
content_path	variable
get_random_img	f()



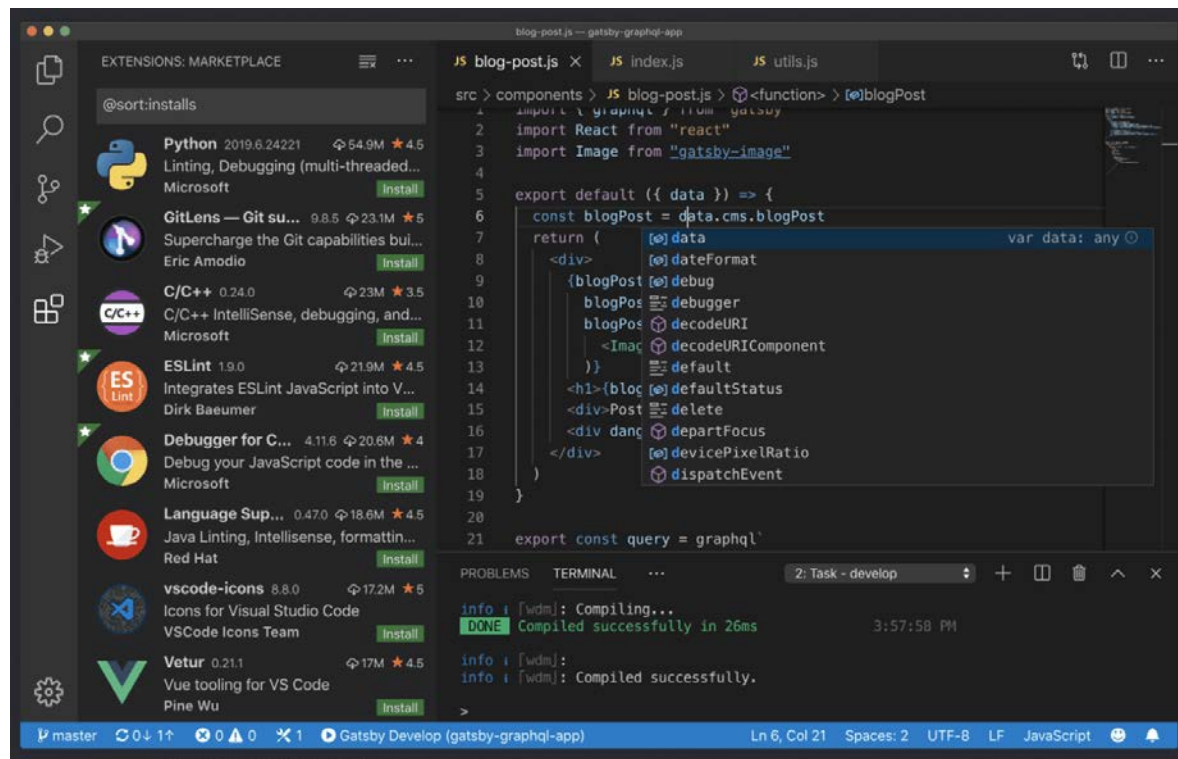
### Prepare the data

Let's create methods that will allow us to load and preprocess our images easily. We perform the same preprocessing expected according to the VGG training process. VGG networks are trained on image with each channel normalized by mean = [103.939, 116.779, 123.68] and with channels BGR.

Deepnote

Бесплатно для личного  
использования и преподавания

# Обзор программ (cont'd)



VS Code

# Заключение

Ограниченные возможности не должны быть равны ограничению в образовании.



**Спасибо за внимание!**