

Time	Narration
00:01	به برنامه آموزشی <b>Advanced matrix operations</b> خوش آمدید.
00:07	در این برنامه یاد می‌گیرید که <b>Frobenius</b> و <b>infinity norm</b> از <b>matrix</b> را بدست آورید.
00:14	<b>singular value decomposition</b> از <b>matrix</b> را پیدا کنید.
00:19	برای ضبط این برنامه من از سیستم عامل
	<b>Ubuntu Linux 16.04</b>
	<b>Python 3.4.3</b> و
00:26	<b>IPython 5.1.0</b>
	استفاده می‌کنم.
00:33	برای تمرین این برنامه شما باید در مورد <b>Lists, arrays</b>
	و دسترسی به بخشی از <b>arrays</b> و انجام دادن <b>matrix operations</b> بدانید.
00:46	اگر نه برنامه آموزشی <b>Python</b> مربوطه در این وب سایت را ببینید.
00:51	ابتدا در مورد <b>flatten function</b> می‌بینیم.
00:55	<b>flatten() function</b> که کپی از <b>array</b> را که به یک <b>dimension</b> تبدیل شده را می‌دهد.
01:01	این را می‌توان برای تبدیل <b>multidimensional matrix</b> به <b>single dimension matrix</b> استفاده کرد.
01:08	<b>ipython</b> را شروع می‌کنیم.
	<b>terminal</b> را باز کنید.
01:13	<b>lpython3</b> را تایپ کنید و <b>Enter</b> را فشار دهید.
	از اینجا به بعد ، به یاد داشته باشید که پس از تایپ هر <b>command</b> در <b>terminal</b> کلید <b>Enter</b> را فشار دهید.
01:25	حالا می‌بینیم چگونه <b>arrays</b> را ایجاد کنیم.
01:29	<b>from numpy import arange</b> را تایپ کنید.
01:35	<b>a</b> مساوی <b>arange</b> داخل براکت 10 <b>reshape</b> 3 <b>dot</b> داخل براکت 3 <b>comma</b>
01:48	<b>a</b> را تایپ کنید.
	حالا <b>flatten open and close brackets</b> را تایپ کنید.
01:57	ابتدا <b>arange function</b> را از <b>numpy module</b> وارد می‌کنیم.
02:02	اینجا می‌بینیم که <b>by 3 matrix 3</b> به <b>one dimensional matrix</b> تبدیل شده است.
02:08	سپس در مورد <b>frobenius norm</b> می‌بینیم.
02:12	این به صورت <b>square root</b> از جمع <b>absolute squares</b> از <b>elements</b> آن می‌باشد.
02:18	ویدیو را موقتاً متوقف کنید.
	این تمرین را امتحان کنید و سپس ویدیو را از سر بگیرید.
02:24	<b>frobenius norm</b> از <b>inverse</b> از <b>by 4 matrix 4</b> داده شده را پیدا کنید.
02:30	به <b>terminal</b> برگردید.

02:34	m مساوی <b>asmatrix</b> داخل براکت <b>arange</b> داخل براکت 1 کما 17 <b>dot reshape</b> داخل براکت 4 کما 4 را تایپ کنید.
02:46	اینجا ما از <b>asmatrix, arange</b> و <b>reshape functions</b> استفاده کرده ایم.
02:52	ما یک <b>matrix</b> با اندازه 4 (در) 4 <b>by</b> که شامل <b>elements</b> از 1 تا 16 است ایجاد کرده ایم.
02:59	حالا <b>m</b> داخل <b>square brackets 0</b> کما 1 مساوی 0 را تایپ کنید.
03:06	<b>m</b> داخل <b>square brackets 1</b> کما 3 مساوی 0
03:12	سپس <b>m</b> را تایپ کنید. ما مقدار <b>element</b> در ردیف 0 ستون 1 و ردیف 1 ستون 3 را به 0 تغییر داده ایم.
03:23	برای پیدا کردن <b>Frobenius norm</b> از <b>inverse</b> از <b>matrix m</b> همانطور که نشان داده شده تایپ کنید.
03:33	<b>norm function</b> در <b>numpy.linalg module</b> در دسترس می باشد.
03:39	سپس در مورد <b>infinity norm</b> از <b>matrix</b> می بینیم.
03:44	این بعنوان مقدار حداکثر (ماکسیموم) از جمع <b>absolute value</b> از <b>elements</b> در هر ردیف تعریف شده است.
03:51	ویدیو را موقتاً متوقف کنید. این تمرین را امتحان کنید و سپس ویدیو را از سر بگیرید
03:57	<b>infinity norm</b> برای <b>matrix im</b> را پیدا کنید.
04:01	برای حل کردن به <b>terminal</b> برگردید.
04:05	برای بدست آوردن <b>Infinity norm</b> برای <b>matrix im</b> همانطور که نشان داده شده تایپ کنید.
04:11	در اینجا مقدار برای <b>ord parameter</b> بعنوان <b>inf</b> برای محاسبه <b>infinity norm</b> پاس شده است.
04:18	برای دانستن بیشتر در مورد <b>norms</b> شما <b>norm question mark</b> را تایپ کنید.
04:24	<b>q</b> را برای خارج شدن فشار دهید.
04:28	سپس در مورد <b>singular value decomposition</b> می بینیم.
04:33	در <b>linear algebra</b> که <b>singular value decomposition</b> فاکتور از <b>real</b> یا <b>complex matrix</b> می باشد.
04:42	<b>SVD</b> از <b>matrix m1</b> را میتوان با استفاده از <b>svd function</b> که در <b>numpy.linalg module</b> می باشد، پیدا کرد.
04:52	همانطور که نشان داده شده تایپ کنید.
04:56	<b>svd</b> که <b>tuple</b> از 3 <b>elements</b> را باز می گرداند.
05:00	ما این مقادیر را در <b>U, sigma</b> variable و <b>V underscore conjugate</b> باز کرده ایم.
05:08	<b>Capital U</b> را تایپ کنید.
05:12	<b>sigma</b> را تایپ کنید.
05:15	<b>Capital V underscore conjugate</b> را تایپ کنید.
05:20	ما میتوانیم <b>singular value decomposition</b> را با مقایسه حاصل <b>U, sigma</b> و <b>V underscore conjugate</b> با <b>m1</b> معتبر کنیم.

05:30	sigma که یک dimensional array است که تنها شامل diagonal elements از matrix می باشد.
05:37	همانطور که نشان داده شده تایپ کنید. ما ابتدا این array را به یک matrix تبدیل می کنیم.
05:43	smat را تایپ کنید. smat یک 2 در 3 (ماتریس) zero matrix می باشد.
05:51	smat داخل square brackets colon 2 comma colon 2 مساوی diag داخل براکت sigma را تایپ کنید.
06:02	سپس smat را تایپ کنید.
06:06	این مقادیر در ردیف 0 ستون 0 و ردیف 1 ستون 1 در smat را با مقادیر از sigma جایگزین می کند.
06:16	smat یک 2by 3 matrix است که برای ضرب کردن با قرار دادن مقادیر از sigma بعنوان diagonal elements و صفر در جاهای دیگر ایجاد شده است.
06:27	همانطور که نشان داده شده است تایپ کنید.
06:33	این True را می دهد. این یعنی elements در m1 و در حاصل U, sigma و V underscore conjugate مساوی هستند.
06:43	به پایان این برنامه می رسیم. خلاصه می کنیم.
06:49	در این برنامه یاد گرفتیم که: norm از یک matrix را با استفاده از () function norm محاسبه کنیم.
06:56	SVD از matrix را با استفاده از () function svd محاسبه کنیم.
07:01	و تمرین برای شما: norm داخل براکت A کاما ord مساوی داخل single quotes fro مثل norm داخل براکت A می باشد درست یا اشتباه
07:19	و پاسخ درست است. چون order مساوی داخل single quotes fro برای Frobenius norm می باشد.
07:29	لطفاً سؤالات زمان بندی شده خود را به این انجمن بفرستید.
07:33	لطفاً سؤالات کلی خود در مورد Python را به این انجمن بفرستید.
07:37	تیم FOSSEE پروژه TBC را هماهنگ می کند.
07:41	بودجه پروژه Spoken Tutorial توسط NMEICT, MHRD, دولت هند تأمین می شود. به این سایت مراجعه کنید.
07:50	ترجمه و صدا گذاری شبنم اقبال از IIT Bombay. با تشکر از شما