Take home exam
Orthogonal polynomials and symbolic computation **Summer 2023**

If below it says "by whichever means you prefer" this refers to using either classical methods or application of symbolic procedures that have been introduced in the lecture. It does not mean citing a solution from the internet, a book, a friend, etc. As a basis for the grades the maximum number of points is 36 and the passing level is 18 points. You can choose freely from the exercises below to achieve these points.

Submission deadline: July 10, 2023.

Exercise 1 (4P) Show that the normal sequences (a) $(x^n)_{n\geq 0}$ and (b) $(x^{\underline{n}})_{n\geq 0}$ are not sequences of orthogonal polynomials with respect to any weight function.

Exercise 2 (4P) Let $k_n(x,y)$ be the kernel polynomials defined from the sequence of polynomials $(\phi_n(x))_{n\geq 0}$ orthogonal with respect to $w(x)$ on $[a,b]$. Show that, if $-\infty < \alpha \leq a < \infty$, then the sequence $(k_n(x,\alpha))_{n\geq 0}$ is orthogonal with respect to the weight function $(x-\alpha)w(x)$.

Exercise 3 (4P) Prove **Theorem 2.5:** For $n,m \geq 0$, $-1 \leq x \leq 1$ and $U_n(x) = \sin(n+1)\theta/\sin\theta$ for $x = \cos\theta$:

(1) $\displaystyle\int_{-1}^{1} U_n(x)U_m(x)\sqrt{1-x^2}\,\mathrm{d}x = 0, \qquad n \neq m$

(2) $U_{n+1}(x) - 2xU_n(x) + U_{n-1}(x) = 0, \qquad n \geq 1$

Exercise 4 (2P for each item) Carry out the details of the proof of **Theorem 2.9**, i.e., show that for $n \geq 0$ and $x \in [-1,1]$ we have:

$$(2n+1)P_n(x) = P'_{n+1}(x) - P'_{n-1}(x) \tag{2.14}$$

$$(n+1)P_n(x) = P'_{n+1}(x) - xP'_n(x) \tag{2.15}$$

$$nP_n(x) = xP'_n(x) - P'_{n-1}(x) \tag{2.16}$$

$$(1-x^2)P'_n(x) = n\left[P_{n-1}(x) - xP_n(x)\right] \tag{2.17}$$

Furthermore, Legendre polynomials satisfy the three term recurrence

$$(n+1)P_{n+1}(x) - (2n+1)xP_n(x) + nP_{n-1}(x) = 0, \tag{2.18}$$

with $P_{-1}(x) = 0$ and $P_0(x) = 1$, and they are a solution to the *Legendre differential equation*

$$(1-x^2)y''(x) - 2xy'(x) + n(n+1)y(x) = 0. \tag{2.19}$$

Exercise 5 (4P) Derive the three term recurrence satisfied by Jacobi polynomials $P_n^{(\alpha,\beta)}(x)$ starting either from Rodrigues formula, Definition 2.11, or the sum representation in Theorem 2.13, by whichever means you prefer.

Exercise 6 (4P) Prove by whichever means you prefer:

(a) $P_n^{(\alpha-1,\beta)}(x) = \frac{n+\alpha+\beta}{2n+\alpha+\beta}P_n^{(\alpha,\beta)}(x) - \frac{n+\beta}{2n+\alpha+\beta}P_{n-1}^{(\alpha,\beta)}(x)$

(b) $P_n^{(\alpha,\beta)}(x) = \frac{n+\alpha}{n+\alpha+\beta}P_n^{(\alpha-1,\beta)}(x) + \frac{n+\beta}{n+\alpha+\beta}P_n^{(\alpha,\beta-1)}(x)$

(c) $(1-x)P_n^{(\alpha+1,\beta)}(x) = \frac{2}{2n+\alpha+\beta+2}\left[(n+\alpha+1)P_n^{(\alpha,\beta)}(x) - (n+1)P_{n+1}^{(\alpha,\beta)}(x)\right]$

(d) $(1-x)\frac{d}{dx}P_n^{(\alpha,\beta)}(x) = \alpha P_n^{(\alpha,\beta)}(x) - (n+\alpha)P_n^{(\alpha-1,\beta+1)}(x)$

Exercise 7 (4P) Let Gegenbauer polynomials for $n \geq 0$ and $\lambda > -\frac{1}{2}$ be defined by the three term recurrence

$$(n+2)C_{n+2}^{\lambda}(x) - 2x(n+\lambda+1)C_{n+1}^{\lambda}(x) + (n+2\lambda)C_n^{\lambda}(x) = 0, \qquad C_0^{\lambda}(x) = 1, \quad C_1^{\lambda}(x) = 2\lambda x.$$

Starting from this definition, compute the generating function $F(z) = \sum_{n\geq 0} C_n^{\lambda}(x)z^n$ by whichever means you prefer.

Exercise 8 (4P) Prove or disprove for $n \geq 1$:

(a) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + T_{n+1}(x)^2 + (1-x^2)U_{n-1}(x)^2 = -x$

(b) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + (1-x^2)U_{n-1}(x)^2 + U_{n+1}(x)^2 = x^2$

(c) $2(1-x^2)T_{n+1}(x)U_{n-1}(x) + T_{n+1}(x)^2 + (1-x^2)U_{n-1}(x)^2 = x^2$

Exercise 9 (4P)

(a) Compute the weighted squared L^2 norm of Hermite polynomials.

(b) Set up the Christoffel-Darboux formula for Hermite polynomials.

(c) Prove Turán's inequality for Hermite polynomials, i.e., show that

$$H_{n+1}^2(x) - H_n(x)H_{n+2}(x) \geq 0, \qquad x \in \mathbb{R},$$

by whatever means you prefer.

Exercise 10 (4P) Implement a procedure in your favourite computer algebra system (CAS) that takes as input a polynomial expanded in the monomial basis and returns its expansion in the basis of falling factorials.

Exercise 11 (4P) Implement a procedure in your favourite CAS that approximates the integral of a given function over the interval $[-1,1]$ using Legendre-Gauß quadrature. The corresponding interpolation points for $n = 7$ are given by

$$\{-0.949108, -0.741531, -0.405845, 0., 0.405845, 0.741531, 0.949108\},$$

and the weights by

$$\{0.129485, 0.279705, 0.38183, 0.417959, 0.38183, 0.279705, 0.129485\}.$$

Test your program at least with a random polynomial of degree 14, $f_1(x) = \mathrm{e}^x$ and $f_2(x) = \tan^2 x - \sin(4x)$.

Exercise 12 (4P) Implement a procedure in your favourite CAS that computes an L^2-approximation up to degree 5 of a given function using the numerical quadrature rule of the previous exercise and include some test cases.

Exercise 13 (4P) Use the Legendre three term recurrence, identity (2.17) and

$$(1-x^2)P_n'(x) = (n+1)\left[xP_n(x) - P_{n+1}(x)\right]$$

to prove that

$$n(n+1)\Delta_n(x) - (n-1)n\Delta_{n-1}(x) = (1-x^2)\left(P_n'(x)P_{n-1}(x) - P_n(x)P_{n-1}'(x)\right), \qquad (3.99)$$

where $\Delta_n(x) = P_n(x)^2 - P_{n-1}(x)P_{n+1}(x)$. Use (3.99) to prove the Turán inequality for Legendre polynomials, i.e., that $\Delta_n(x) \geq 0$ for $n \geq 0, x \in [-1,1]$.

Exercise 14 (4P) Show that for nonnegative x, y, and z one has the implication

$$1 \leq xyz \quad \Longrightarrow \quad 8 \leq (1+x)(1+y)(1+z).$$

In addition show that 8 is optimal on the RHS, by replacing it by a free variable and running CAD to show that it is indeed the largest possible lower bound. Then also replace the 1 on the LHS by a free variable and determine conditions on the two lower bounds for the implication to hold.

Exercise 15 (4P) Compute a CAD for

$$A = \{p_1(x,y) = x^2 + y^2 - 1, p_2(x,y) = x - y\}$$

with the Brown-McCallum projection operator as defined in the lecture:

$$P(A_k) = \bigcup_{p \in A_k} \{\mathrm{lc}_{x_k}(p), \mathrm{disc}_{x_k}(p) \cup \bigcup_{p,q \in A_k} \{\mathrm{res}_{x_k}(p,q)\},$$

using, e.g., Mathematica built-in functions for resultant, etc. on your favorite computer algebra system. Visualize the result graphically. How many cells does the original AD of A have and how many the final CAD?